Practical Electronics

The UK's premier electronics, computing and maker magazine

Max's Cool Beans
Making automata & building the VFD clock

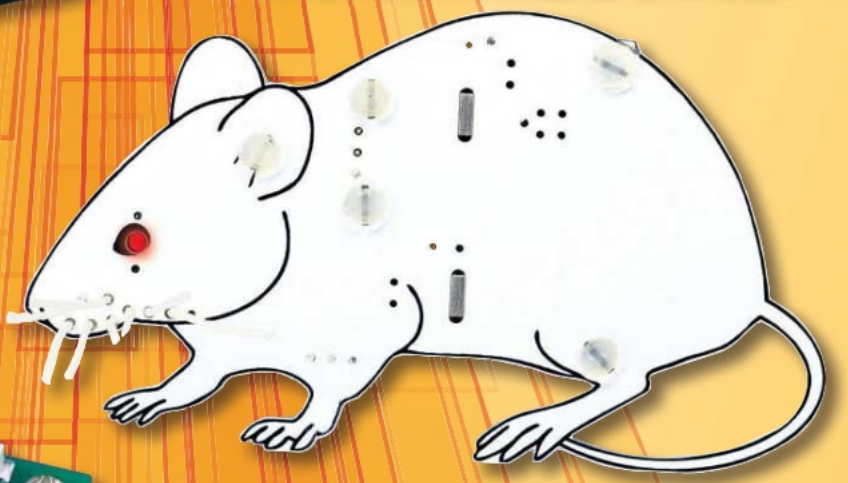
Teach-In 12
Part 11: long-distance radio communications

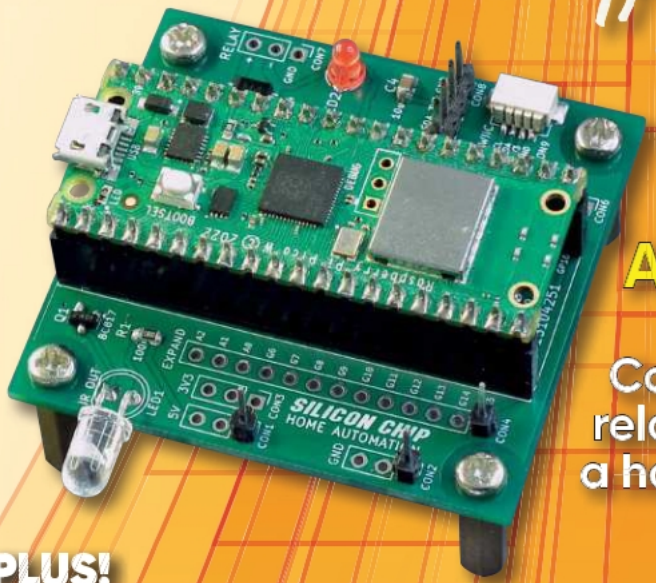
Audio Out
Visiting Retro-Tech 2026

Circuit Surgery
Measuring resistance with a 555 timer chip

Building the USB-C Power Monitor

Mic the Mouse:
a sound-sensitive pet that won't make a mess





Home Automation Satellite
Connect sensors, relays and more to a home automation system

PLUS!

Home Automation – running HomeAssistant on a Raspberry Pi

Power Electronics part six – how DC-to-AC converters work

Techno Talk – display technology changes over the last 50 years

Sep 2026 £5.99

9 772632 573030 09

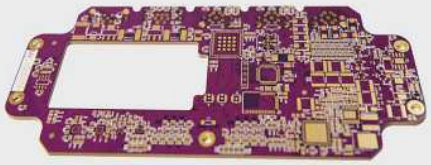
PCBWay

Beyond Standard —

Discover PCBWay's Advanced PCB Lineup

About PCBWay:

With dedicated workshops for advanced PCBs, PCBWay offers full-spec, high-precision, and large-scale production. We ensure customers are supplied with PCBs that meet stringent quality standards at the lowest cost.

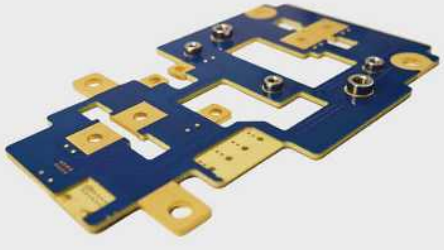


HDI PCB

- 4-60 layers
- FR-4, High Tg, Halogen-free, PTFE
- Finished Copper: 1/3OZ–8OZ
- Min Track/Spacing (inner): 2/2mil (H/HOZ base copper)
- Min BGA Pitch: 0.35mm

High Frequency PCB

- RO4003C, RO4350B, Ro3003, Ro3010, RT5880
- Min. Track/Spacing: 2mil
- Min. Hole Size: 0.15mm
- Finished Cooper: 1-2oz
- Board Thickness: 0.2–3.2mm



Other PCBs with Special Process

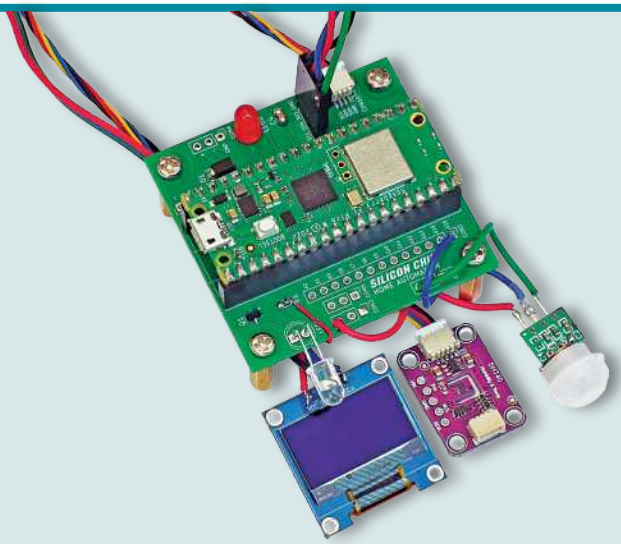
- Embedded Copper PCB
- Backplane PCB
- Cavity PCB
- White/Partially Hybrid PCB
- Black Core PCB (After-dark)

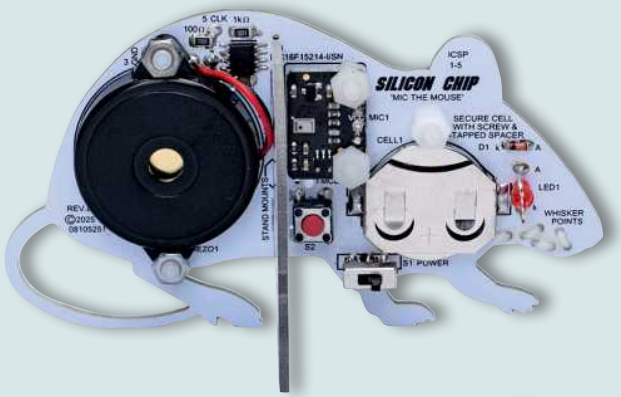
www.pcbway.com service@pcbway.com

Practical Electronics

Volume 55. No. 09
September 2026
ISSN 2632 573X

Contents





Projects and Circuits

Home Automation Satellite by Richard Palmer 10
In a home automation system, a satellite is a device that can carry one or more sensors, displays, relays or LEDs to be controlled by the system. This DIY satellite is straightforward to build and can be fully controlled using HomeAssistant.

Mic the Mouse by John Clarke 26
Mice make good pets but you still have to take care of them. This little critter doesn't eat much (just the occasional lithium cell) and won't make a mess or escape from its cage. Mic is great for playing pranks on friends and family, too.

USB-C Power Monitor, part two by Tim Blythman 68
This compact device lets you monitor the voltage, current, power and energy supplied to a USB-C device. The first article last month primarily covered the circuit details and operation. Now we'll describe the construction and usage of this handy tool.

Series, Features and Columns

Using Home Assistant, part one by Richard Palmer 4
There are many situations around the home and on the workbench where remote sensing and control can improve our lives. HomeAssistant is an excellent DIY home automation system and you can run it on an inexpensive Raspberry Pi board.

Max's Cool Beans by Max The Magnificent 17
Weird & Wonderful Arduino Projects, Part 21: Ant automata antics and testing times We find another use for our Retro Games Console: running celluar automata simulations and such. We also make progress in building our Chronological Contraption.

Audio Out by Jake Rothman 34
The RetroTech UK event and radio rallies, part one of two. RetroTech UK is run by the British Vintage Wireless Society (BVWS). It's a wonderful sale of old electronic stuff that takes place every May. I saw many interesting things when I attended this year.

Power Electronics, part six by Andrew Levido 40
Having covered DC-DC and AC-DC converters in earlier articles in this series, we will now move on to DC-AC converters. They have been around for a long time, but their usage has become widespread over the last couple of decades.

Teach-In 12.11: the World of Wireless by Mike Tooley 46
Long-distance radio communications: this month, we bring our Teach-In to a close by providing you with some ideas that will take you further into the world of radio and wireless technology. Our final Hands-On project is a high-gain RF preselector.

Circuit Surgery by Ian Bell 58
This month we make a diversion into an unusual topic: making music using bio-feedback from plants. That may sound 'out there' but it involves some practical techniques for measuring high resistances using the famous 555 timer IC.

Techno Talk by Max the Magnificent 67, 75
From clattering split-flap displays and glowing VFDs to holographic smart glasses, liquid-cooled smartphones, and virtual acoustic worlds; what might tomorrow bring?

Regulars and Services

Editorial 2
Through-hole parts are not obsolete, but they're getting close

Reader services – Editorial and Advertising Departments 2

Subscribe to Practical Electronics and save money 3

Practical Electronics PCB Service 76-78
PCBs for Practical Electronics projects

Classified ads and Advertiser index 79

Next month! – highlights of our next issue of *Practical Electronics* 80

Made in Australia & the UK.

Written in Britain, Australia, the USA and Ireland.

Read everywhere.

© Silicon Chip Publications Pty Ltd 2026

Copyright in all drawings, photographs, articles, technical designs, software and intellectual property published in *Practical Electronics* is fully protected, and reproduction or imitation in whole or in part are expressly forbidden.

The October 2026 issue of *Practical Electronics* will be published on Friday, 5th of September, 2026 (see p80).

Renewals paid by PayPal, Visa and Mastercard are now handled via our website: www.electronpublishing.com

Practical Electronics

Volume 55. No. 08
August 2026
ISSN 2632 573X

Editorial

Editorial offices
Electron Publishing (Australia)
Tel +61 2 9939 3295
Email pe@pemag.au
Web www.electronpublishing.com

Address mail to:
Electron Publishing (Australia)
PO Box 194, Matraville NSW 2036
Australia

Advertising enquiries
+61 2 9939 3295 pe@pemag.au

Editor	Nicholas Vinen
Publisher	Nicholas Vinen
Digital subscriptions	Stewart Kearn **Tel 07918 614662**
Online Editor	Alan Winstanley
Web Systems	Kris Thain
Production	Bao Smith
Technical staff	Tim Blythman, John Clarke

Print subscriptions
Practical Electronics Subscriptions
PO Box 6337
Bournemouth BH1 9EH **Tel** 01202 087631
United Kingdom **Email** pesubs@selectps.com

Technical enquiries
We regret that technical enquiries cannot be answered over the telephone. We are unable to offer any advice on the use, purchase, repair or modification of commercial equipment or the incorporation or modification of designs published in the magazine.

Questions about articles or projects should be sent to the editor by email: pe@pemag.au

Projects and circuits
All reasonable precautions are taken to ensure that the advice and data given to readers is reliable. We cannot, however, guarantee it and we cannot accept legal responsibility for it.

Some projects and circuits published in *Practical Electronics* employ voltages that can be lethal. Do not build, test, modify or fix any mains-powered equipment unless you fully understand the safety aspects involved and you use an RCD (GFCI) adaptor.

Component supplies
Silicon Chip Publications may offer kits or other parts for making our projects, but not in all cases. When kits are not available, readers will need to find and source parts themselves. We advise readers to check that all parts are still available before commencing any project in a back-dated issue.

Advertisements
Although the proprietors and staff of *Practical Electronics* take reasonable precautions to protect the interests of readers by ensuring as far as practicable that advertisements are bona fide, the magazine and its publishers cannot give any undertakings in respect of statements or claims made by advertisers, whether these advertisements are printed as part of the magazine, or in inserts. The Publishers regret that under no circumstances will the magazine accept liability for non-receipt of goods ordered, or for late delivery, or for faults in manufacture.

Transmitters/bugs/telephone equipment
We advise readers that certain items of radio transmitting and telephone equipment which may be advertised in our pages cannot be legally used in the UK. Readers should check the law before buying any transmitting or telephone equipment, as a fine, confiscation of equipment and/or imprisonment can result from illegal use or ownership. The laws vary from country to country; readers should check local laws.

Through-hole parts are not obsolete, but they're getting close

I frequently get emails asking us to use through-hole parts in our projects, because they're easier to solder (which is fair enough). However, that is becoming increasingly difficult.

While I'm sure through-hole parts will be available for a while longer (you still see them in some cheap new equipment), the range of TH parts that is available has been shrinking for a while now.

Apart from a few specific manufacturers, almost all new integrated circuits are being made available only in SMD packages. In some cases, they are only available in packages like DFN and BGA that aren't intended to be hand soldered (it is possible, but it isn't easy!).

That means if you insist on sticking entirely with through-hole parts, you're largely stuck with late-1990s technology, if not earlier (many TH chips available now were desigend in the 1960s & 1970s).

Op amps are a case in point. After Jacob Rothman pointed out that the NE5532 was being replaced with an inferior version, I decided to look at what other good op amps were available in through-hole DIP packages. The answer is: not very many, and mostly older ones, like the LM358 and TL072. In a few years, if you want to buy a good, new audio op amp, you'll have to get it in a package like SOIC-8 or something even smaller like MSOP.

So designing new circuits using only through-hole ICs now imposes significant limitations on component choice. This is particularly evident in entire classes of devices, where modern through-hole options are rare to non-existent, including:

- High-resolution ADCs and DACs
- Mosfet gate drivers
- Microcontrollers (AVRs and PICs are the main exception, but even then, the through-hole parts are a subset of what's available)
- Switching regulators
- Many sensor types, especially MEMS devices
- High-speed op amps and specialised RF devices

There's also the size consideration. These days, to make our designs competitive with commercial equivalents, we need to fit them into convenient and attractive enclosures. That's more difficult when the PCB is several times the size.

We aren't going to stop publishing designs using primarily through-hole parts any time soon. But the trend you will have noticed – with an increasing number of projects using some SMDs, or even exclusively smaller parts – is simply us following industry trends.

After all, our projects are meant to be educational, and that includes learning to work with the parts used in most commercial products today.

Nicholas Vinen, Electron Publishing (Australia)*
* a division of Silicon Chip Publications Pty Ltd.

Practical Electronics

UK readers SAVE £1 on every issue

SUBSCRIBE NOW!

01202 087631

Take out a one-year subscription and save more than £10 over the year.
Even better – save over £1 per issue if you subscribe for two years – a total saving of £26.

Overseas rates represent exceptional value

You also:
- **Avoid any cover price increase for the duration of your subscription**
- **Get your magazine delivered to your door each month**
- **Guarantee a copy, even if your local newsagent sells out**

Order by:
- **Phone or post with a cheque, postal order or credit card**
- **Or order online at: www.electronpublishing.com**

Practical Electronics

SUBSCRIPTION PRICES

Subscriptions for delivery direct to any address in:

UK: 6-months £33.99, 12-months £59.99, 24-months £114.99

Europe Airmail: 6-months £38.99, 12-months £69.99, 24-months £129.99

Rest Of The World Airmail: 6-months £46.99, 12-months £84.99, 24-months £164.99

Cheques (in **£ sterling only**) payable to ***Select Publisher Services Ltd*** and sent to: Practical Electronics Subscriptions, PO Box 6337, Bournemouth BH1 9EH, United Kingdom

Tel: 01202 087631
Email: pesubs@selectps.com

Also via our online shop at**: www.electronpublishing.com**

Subscriptions start with the next available issue. We accept MasterCard or Visa.
(For past issues see the Back Issues page.)

DIGITAL SUBSCRIPTIONS

Electronic subscriptions are available from £29.99 for 12 months, more details at: **www.electronpublishing.com**

SUBSCRIPTION ORDER FORM

☐ **6 Months:** UK £33.99, Europe £38.99 (Airmail), Rest of the World £46.99 (Airmail)

☐ **1 Year:** UK £59.99, Europe £69.99 (Airmail), Rest of the World £84.99 (Airmail)

☐ **2 Years:** UK £114.99, Europe £129.99 (Airmail), Rest of the World £164.99 (Airmail)

To: Practical Electronics Subscriptions
PO Box 6337
Bournemouth BH1 9EH
United Kingdom

Tel: 01202 087631

Email: pesubs@selectps.com

I enclose payment of £ .....
(cheque/PO in £ sterling only)
payable to ***Select Publisher Services Ltd***

☐ Please charge my Visa/Mastercard

My card number is: .....
Please print clearly, and check that you have the number correct

Card Security Code ..... Valid From Date.....
(The last 3 digits on or just under the signature strip)

Card Ex. Date .....

Name .....

Address .....

Post code ..... Tel. .....

Email .....

Subscriptions start with the next available issue.

Part 1 by Richard palmer

HOME ASSISTANT

WITH A RASPBERRY PI

There are many situations around the home and on the workbench where remote sensing and control can improve our lives.

This short series will explore the world of home automation and create the core of a system that interfaces with a broad range of commercial smart home devices, plus hundreds of DIY sensors and remote-control interfaces. That will include everything from sprinklers to music systems, pet tags to air conditioners.

Home automation has matured to the point where do-it-yourself installations can be achieved without needing to write code. Readily available sensors and controls can be wirelessly connected to a central hub and added to the system with just a little configuration information.

This month, we will review some community-supported home automation platforms. We'll also discuss our simple project (in this issue) that can connect sensing and control devices to a central home automation hub using a WiFi network.

A follow-up article next month will have information on more advanced matters, such as remote access, customised dashboards and cameras.

Smart homes & IoT

IoT (the Internet of Things) connects the physical and virtual worlds using sensors and controls connected to monitoring and automation software via communication networks. Home automation is a subset of IoT focused on the domestic environment.

Home automation begins with sensing something in the environment, such as time, temperature, light intensity, the presence of smoke, or movement. A set of rules is applied to automate actions, such as switching a light on or off, or sending a notification based on sensed changes.

A typical home automation system includes an in-home server (the 'hub'), and a range of 'satellite' devices distributed around the property, communicating using some form of wireless link.

That link could be a local WiFi network, Bluetooth or a Zigbee mesh. The hub is either a stand-alone device, or connects to a cloud-based platform which enables remote access – see Fig.1.

There are a multitude of commercially available devices that can directly interact with home automation services. A quick internet search turned up 940 items on the Currys website and 829 listed by Argos, covering lighting, gardens, security and even smart pet-tech.

Similar searches via AliExpress and eBay each returned more than fifty pages of results.

Where devices are not designed to be directly controlled by such a system, automated control can often be implemented by switching their power on and off, or mimicking an existing control capability, such as an infrared (IR) remote. For devices with no remote control capabilities, there are 'fingerbots' that can press a button on command!

Choosing a platform

There are excellent and easy-to-use commercial integration platforms such as Google Assistant, Amazon Alexa,

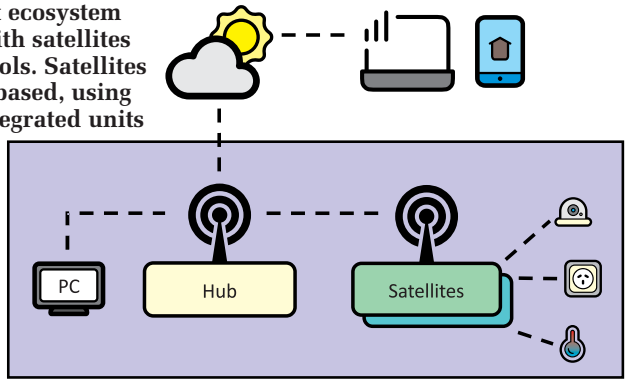
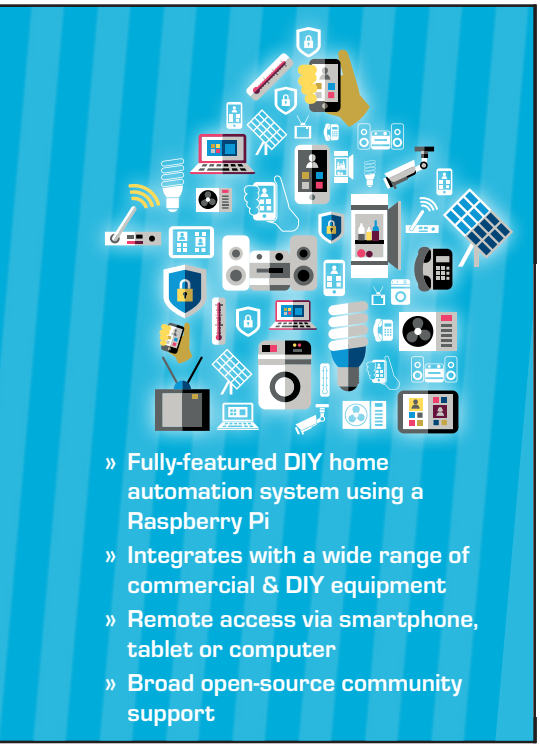


Fig.1: the HomeAssistant ecosystem comprises a local hub with satellites that host sensors & controls. Satellites may be microcontroller-based, using ESPHome, or be fully integrated units such as IP cameras. Communication between elements commonly uses WiFi, Bluetooth or Zigbee. Remote access to the system can be provided by a VPN or cloud service.



- Fully-featured DIY home automation system using a Raspberry Pi
- Integrates with a wide range of commercial & DIY equipment
- Remote access via smartphone, tablet or computer
- Broad open-source community support

Apple HomeKit, Philips Hue and Samsung SmartThings.

Each has its pros and cons. However, these platforms are largely devoid of features that support DIY. Almost all require a commercial in-home hub and an online account.

Mix-and-match integration of various vendor's 'ecospheres' into one master platform can be a frustrating and sometimes unattainable goal.

Fortunately, there are several groups of enthusiasts that have created open-source home automation ecosystems that can integrate with a range of commercial hardware and also support DIY applications. Two enthusiast-friendly platforms stood out as I began researching this series: openHAB and HomeAssistant.

They both offer a fully featured hub, integrations with a wide range of DIY and commercial home automation products, remote access and integrated cloud services. Both host their hubs on the Raspberry Pi platform using customised operating systems.

They both offer voice control via integration with smart home assistants such as Google Assistant, Siri or Alexa.

For DIY projects, HomeAssistant (HA) stood out. From the hub's browser interface, you can configure and manage remote sensors and controllers based on WiFi-capable microcontrollers such as the Raspberry Pi Pico W and Espressif Systems wireless microcontrollers (eg, the ESP8266 & ESP32) using HA's ESPHome firmware. More than a hundred sensors and control interfaces are pre-integrated.

While HomeAssistant's cloud service costs around £5/month, remote access can be configured at no cost using one of several free VPN services.

openHAB also has a very wide range of integrations with commercial home automation devices; however, integrating DIY projects is more complex. Its cloud platform, myopenHAB, is free.

While openHAB's free cloud service was very tempting, we selected HomeAssistant for this series based on its flexible DIY device capabilities.

Installation on a Raspberry Pi 3B, 4 or 5 is straightforward, or it comes pre-installed on HA's Yellow or Green platforms.

HomeAssistant

HomeAssistant has three layers: a basic Linux operating system, the HA Core that interacts with users, devices and services and the Supervisor, which orchestrates the various HA components and manages backups and updates – see Fig.2.

Home Assistant's language can be confusing at first. A few key definitions may be helpful:

- An 'entity' is the basic building block of home automation. It represents a single sensor, control element or function. Entities have 'states', which may be binary or a range of values.
- A 'device' may host a single entity, such as a switch or a light sensor, or several entities, such as temperature, pressure and humidity sensors.
- An 'area' is a logical grouping of entities and devices, often representing a geographic location, such as a kitchen. Areas can be assigned to 'floors'.
- An 'integration' is the software that connects HomeAssistant to a compatible device, such as an ESPHome satellite or a webcam, or another home automation or service platform such as email or the HomeAssistant Cloud.
- 'Automations' are sets of repeatable actions that can run automatically. They comprise 'trigger' events, tests for 'conditions' on those events and resulting 'actions'.

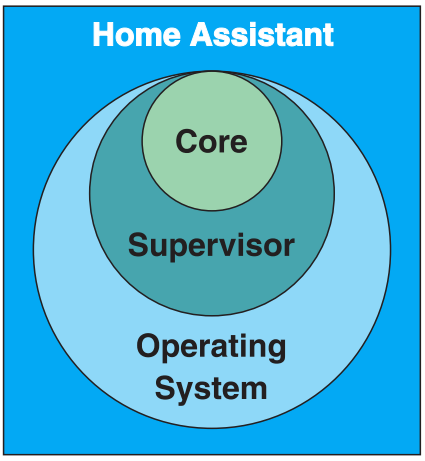
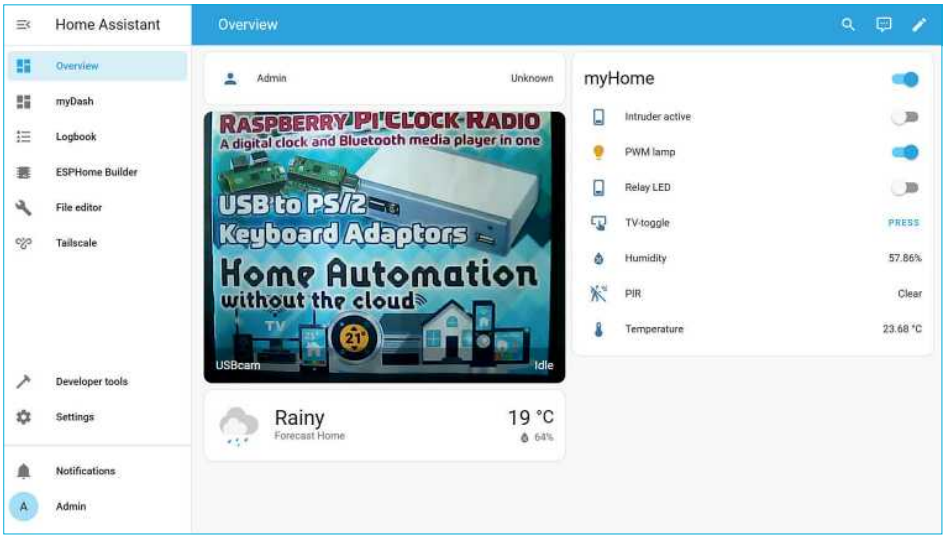


Fig.2: the HomeAssistant platform has three layers: a basic Linux operating system, a Supervisor layer and the HA Core. *Source:* *https://developers.home-assistant.io/docs/architecture_index*



The HomeAssistant Overview dashboard with the ESPHome satellite and a USB webcam installed.

Another practical application

While writing this article, a local community radio station needed to improve the monitoring and control of the equipment at its transmitter site from the studios, or remotely using the engineering team's laptops or phones.

The station's two transmitters, uninterruptible power supply (UPS), studio-transmitter link and backup program links have a range of digital and analog inputs and outputs for monitoring and control.

HomeAssistant and ESPHome seemed a good solution to their problem. Pico-based satellite hardware like that described in the accompanying project article was used for the thirty-odd digital and analog values to be monitored and drive relay and opto-isolated outputs.

It only took a few hours to integrate the hardware into ESPHome and create the required sensors and controls in HomeAssistant. No custom code was required, saving weeks of programming and improving reliability.

This fingerbot can press a button on any device as a simple remote control.

You can find a more detailed discussion of these concepts at https://pemag.au/link/abr3

The Home Assistant hub

We chose to create our own hub using a Raspberry Pi. Any Raspberry Pi, from the Model 3B onwards, will do. All that is required besides the Pi is a 32GB Class A2 microSD card for the custom HAOS operating system. An existing Raspbian SD card can be overwritten with the new OS.

Details of the requirements and options are available on the HomeAssistant website (https://pemag.au/link/ac5w).

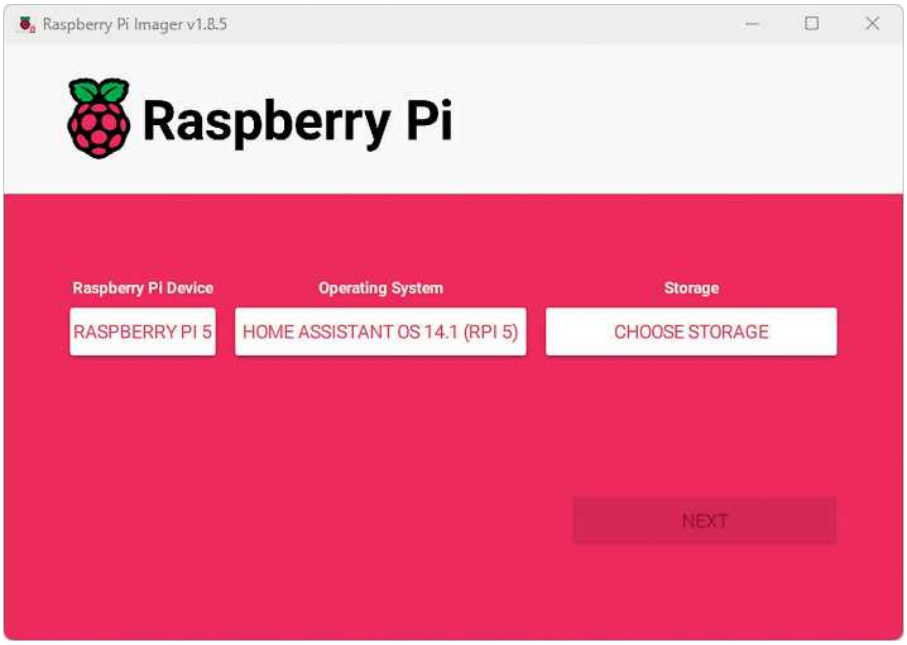
While many home automation needs can be satisfied with off-the-shelf commercial components, there are still situations where it is more practical and less expensive to create a DIY solution.

There are also situations where a compatible remote control device isn't available commercially, but the communication protocol is well-documented. TVs and air conditioners are a case in point.

To assist with these needs, we have developed a small PCB that can connect to a wide range of digital, analog and I²C sensors, as well as controlling devices via an IR signal or relay. It is presented as a separate project in this issue, starting on page 54.

Setting up HomeAssistant

For the purposes of this tutorial, we will use a Raspberry Pi Model 5 as the hub. However, the process is similar for other platforms. Detailed instructions are available at https://pemag.au/link/ac5w



Screen 1: when preparing the SD card, select Home Assistant OS in the Raspberry Pi Imager app.

When procuring a case for the Raspberry Pi, avoid those made entirely of metal as they tend to reduce the WiFi range significantly. The official red-and-white plastic case or a clear acrylic case are solid choices. The active cooler kit is also a good investment and fits neatly into the acrylic case.

The HomeAssistant and ESPHome communities are very active, producing updates several times a month. For this reason, some of the instructions below may be out of date by the time this series is in print. Don't despair if this happens; the documentation is kept up to date and help is available via the very responsive HomeAssistant user community (https://community.home-assistant.io).

If you don't want to build your own hub, the pre-configured HomeAssistant Green hub is readily available for around the same price as a well-configured Raspberry Pi 5 and matching case.

Preparing the Pi is straightforward. If it came in kit form, just fit the motherboard into its case and mount the cooler's heatsink. The fan cable goes into the connector behind the outside-edge USB connectors. There is no need to load or configure the Raspbian OS, as it will be replaced by HA's HAOS operating system.

Install and run Raspberry Pi's Imager software on a Windows, Mac or Linux PC – see Screen 1. Select your Raspberry Pi board from the list in the first box, then choose the required operating system in the second box, by selecting "Other specific-purpose OS", then "Home Assistants and Home Automation", then "Home

Assistant" and the blue-logo version from the drop-down menus.

Plug in an SD card adaptor with the microSD card installed (or, if your computer/monitor has an integrated SD card socket, use that). Select that card in the final Imager box. Click NEXT, and the card will be programmed over several minutes after an erasure warning box is displayed.

Installing the OS

There is no need for a screen, keyboard and mouse for the Pi as the installation is 'headless'. Wired Ethernet is mandatory for the initial installation process, but it can be replaced by WiFi once HA is configured.

Insert the SD card into the Pi and an Ethernet cable between the Pi and your WiFi router. Connect the power adaptor and press the power button on the Pi if it remains red after connection. Wait for 4-5 minutes for HAOS to initialise. The green power LED will flash intermittently during this time.

The final configuration steps are completed via a browser on your PC; enter "http://homeassistant.local:8123/" in the URL bar. Refresh the URL regularly until the HomeAssistant welcome page loads.

Click on the "Create my smart home" button. The Name field is used for display, while the Username is used for logging in. They can be the same name. Add your location, which will be used for the weather displays and proximity mapping of any devices you choose to track. Select what information you wish to share with HomeAssistant's developers.

A screen of compatible devices that HomeAssistant has located on your network completes the basic installation process. The HomeAssistant Overview screen should then appear, as shown in Screen 2.

Click on the Settings menu item and then Network. Under "Configure network interfaces", click on WLAN0 and SEARCH NETWORKS. Select your WiFi network from the list, choose the appropriate security scheme, and enter your WiFi password. Click the SAVE button at the bottom of the block, not the one at the bottom of the page.

Now that your WiFi credentials have been entered, the unit no longer requires a wired connection.

HomeAssistant-compatible cameras

Several types of cameras can be added to HomeAssistant. The simplest is a USB webcam connected directly to the hub. Most commercial USB webcams should be compatible.

If a remote camera is required, ESP32-based ESP-CAMs are available at very low cost and with acceptable picture quality. A USB-CAM with the standard OV2640 camera, including the USB adaptor board, costs less than $10 on AliExpress (eg, AliExpress 1005006501528278).

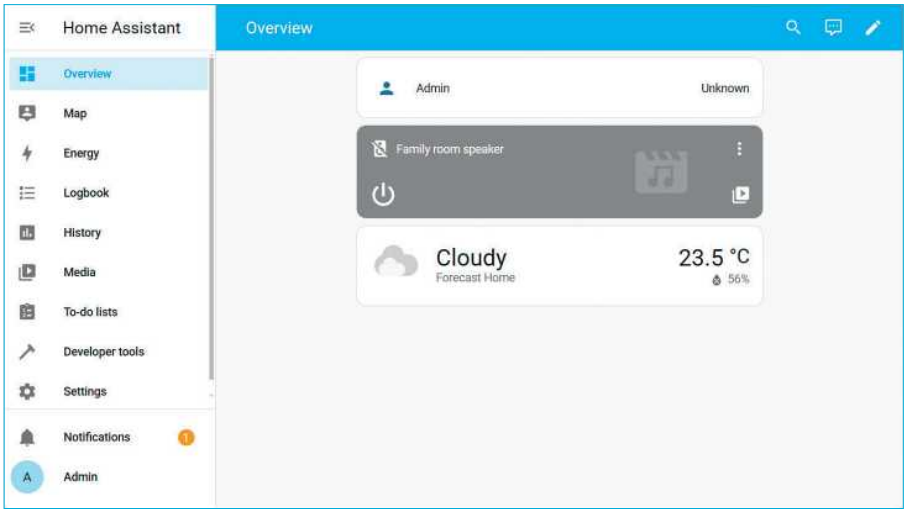
While the OV5640 version provides a higher resolution image, plus autofocus on models with the silver lens surround, I do not recommend it. The one we tested had marginal low-light performance and tended to overheat, reaching temperatures of 70°C within a few minutes. A heatsink attached to the back of the sensor reduced the temperature to 40°C.

Commercial WiFi cameras offer more advanced features such as pan and tilt, automatic IR illumination at low light levels, motion tracking and substantially better images. However, they are the most difficult to integrate due to a wide variety of firmware platforms being used.

Of the WiFi cameras I tested, those using the V380 or V380 Pro configuration apps and marked ONVIF compliant in their specifications were the most likely to be compatible.

We'll have more on using cameras with HomeAssistant in the follow-up article next month.

This £5 USB webcam produced very acceptable images.



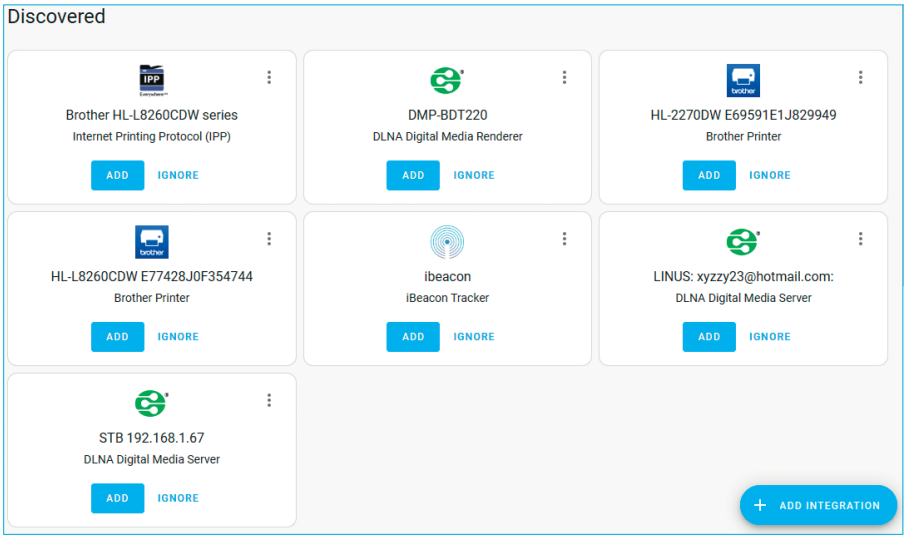
Screen 2: the Overview dashboard after the system has been initialised. The auto-discovery process located a Google home device; local weather is displayed by default.

Parts List – Home Assistant

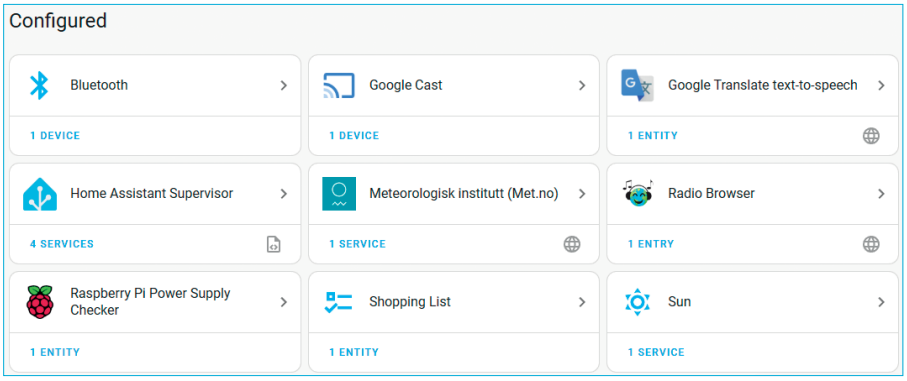
1 Raspberry Pi Model 3B or greater, with at least 4GB of RAM
1 Power supply for the Raspberry Pi
1 non-metallic case
1 active cooler kit for the Raspberry Pi
1 32GB Class A2 microSD card
1 USB microSD card reader/writer (if your computer doesn't have one)
1 Ethernet cable (for installation)

Left: a $10 ESP32-CAM with an OV2640 sensor.

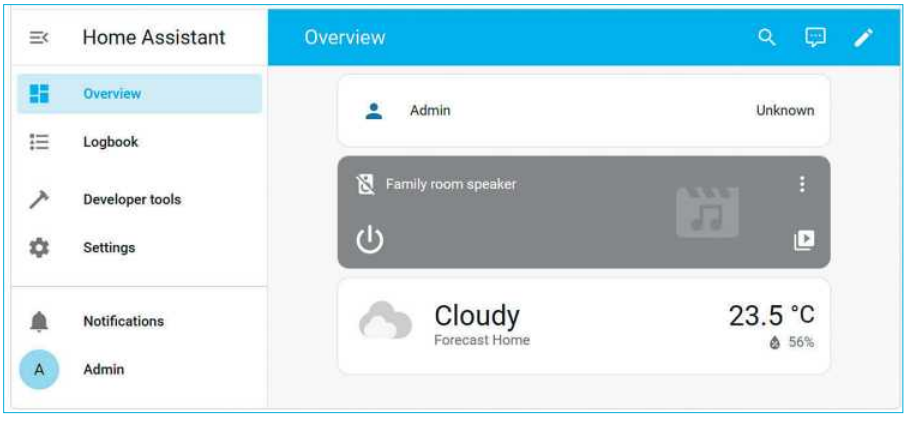
Right: a $50 WiFi camera with pan and tilt functions and IR-illuminated night vision. ONVIF compliance is essential for HA compatibility.



Screen 3: HomeAssistant automatically scans the network for compatible devices.



Screen 4: some services are enabled by default.



Screen 5: the simplified HA menu after hiding some items.

The username and password you set may also be used for smartphone, tablet and remote browser access.

It is a good idea to use credentials without full administrator privileges whenever possible to reduce the possibility of accidental misconfiguration, and to increase remote access security.

The basic installation is now complete, and you can start configuring your smart home. Before adding any new devices, it is useful to clean up the devices that have been auto-discovered.

Select Settings near the bottom of the left-hand menu and click on Devices & services. A screen of discovered devices will appear, like in Screen 3. Devices like wireless-capable printers and set-top boxes will be included. Click IGNORE on all except the iBeacon Tracker and perhaps a printer. They will clutter up the screen and can be easily re-discovered later.

Further down the screen is a list of configured services (see Screen 4). There are some generic services, such as Bluetooth and the Home Assistant Supervisor, as well as some applications like a live-streamed radio channel browser and a weather service from the Norwegian Meteorological Institute.

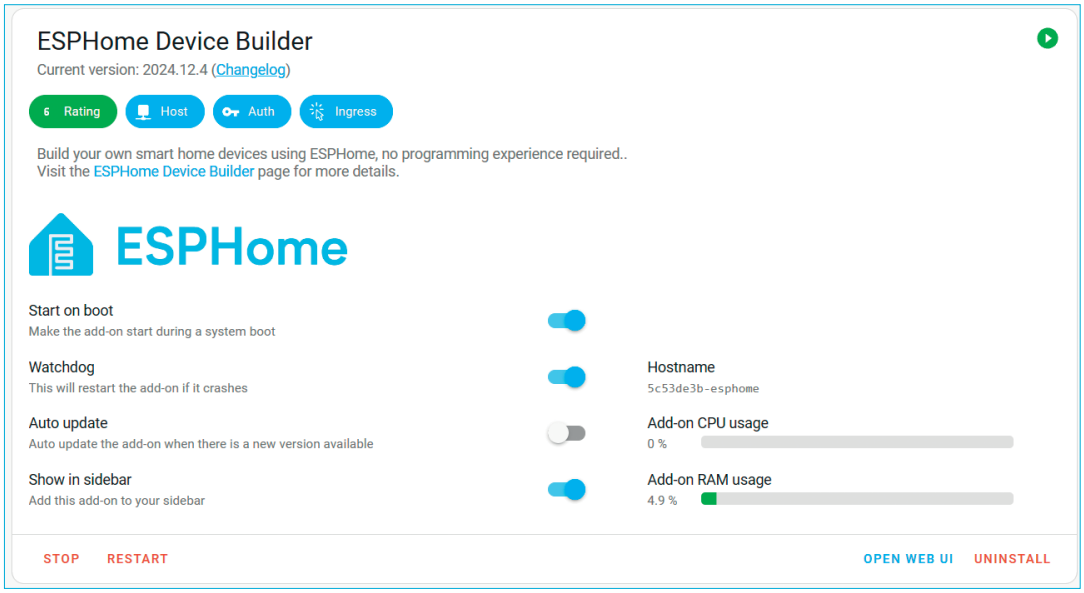
I clicked on “Shopping list”, then the three dots at the side of the “Integration entries” panel and then Disable to hide that service.

The sidebar menu

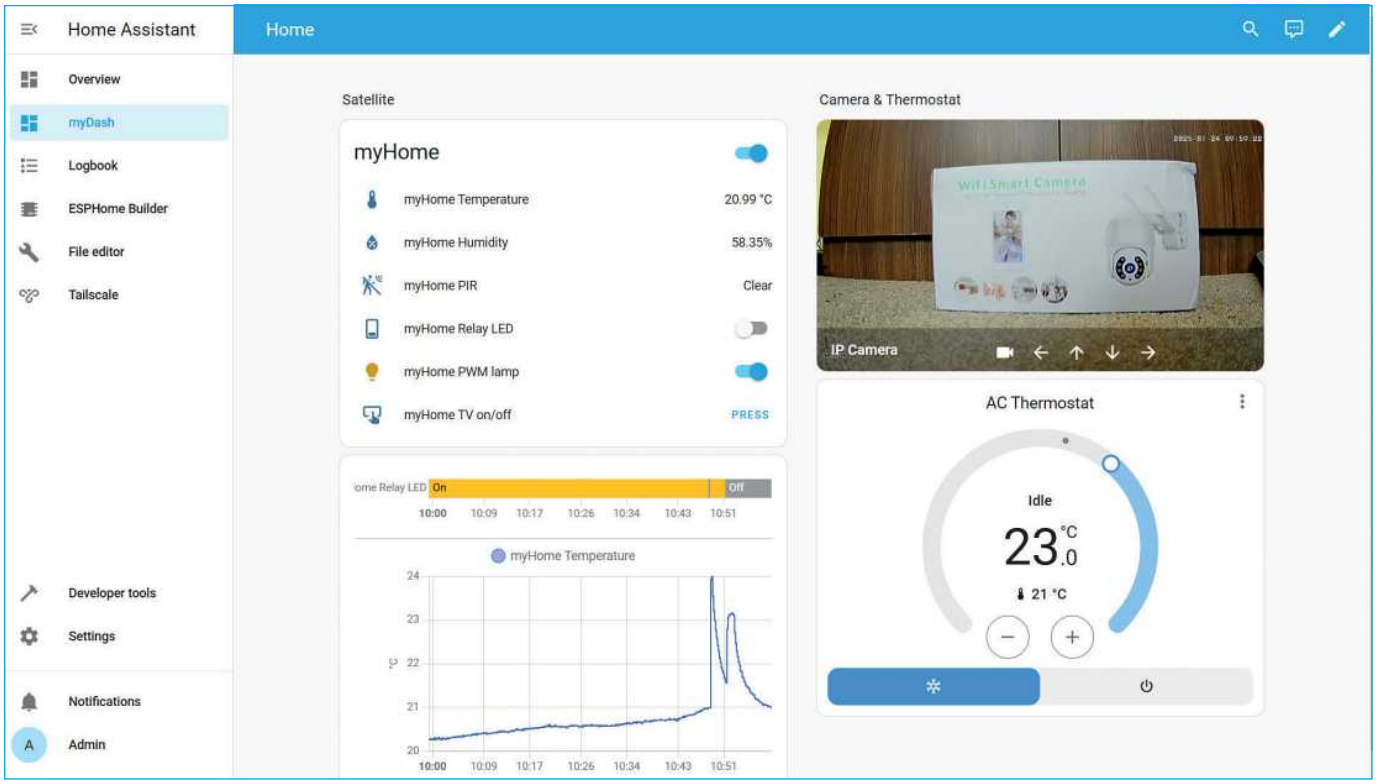
The sidebar menu has some items you may rarely use. To customise what’s displayed, click on your username at the bottom of the sidebar. Scroll down to the “Browser settings” block and click EDIT to change the order and hide items from the sidebar. Click on the X next to the items you don’t want to appear. The order can be changed by sliding the items to your desired location.

Click DONE at the top of the main menu bar when you are finished. Initially, I hid all the optional items other than the Overview dashboard and the Logbook – see Screen 5.

The “ESPHome” add-on will be needed to add the satellite device that’s described in the separate project article in this issue (it won’t hurt to add it regardless). Go to Settings then Add-ons, click on the ADD-ON



Screen 6: configuring the ESPHome Device Builder.



Screen 7: a custom HomeAssistant dashboard showing the satellite's sensors and controls, an IP camera's image and a thermostat automation. The spikes in the temperature reading are from a finger being placed on the sensor. The thermostat temperature was raised during the second spike.

STORE button at the bottom right, search for "ESPHome". Select it and click the INSTALL link at the bottom of the tile, and then START once the installation is complete.

Before exiting the installation menu, enable the "Show in sidebar" and "Watchdog" options – see Screen 6. ESPHome Builder should now appear in the left sidebar menu. Select it.

Click on the SECRETS button at the top right-hand corner of the ESPHome tab. The edit window should contain something like the following:

```
## secrets.yaml
wifi_ssid:
  "your wifi ssid"
wifi_password:
  "your wifi password"
```

Fill in your network credentials and save them. This will allow new ESPHome devices to be automatically configured for your network. Click on the X next to the file name to close the edit window.

Conclusion

A separate project in this issue covers the construction of a satellite board based on a Pico W. Among other things, it supports temperature, humidity and motion sensors and an OLED display.

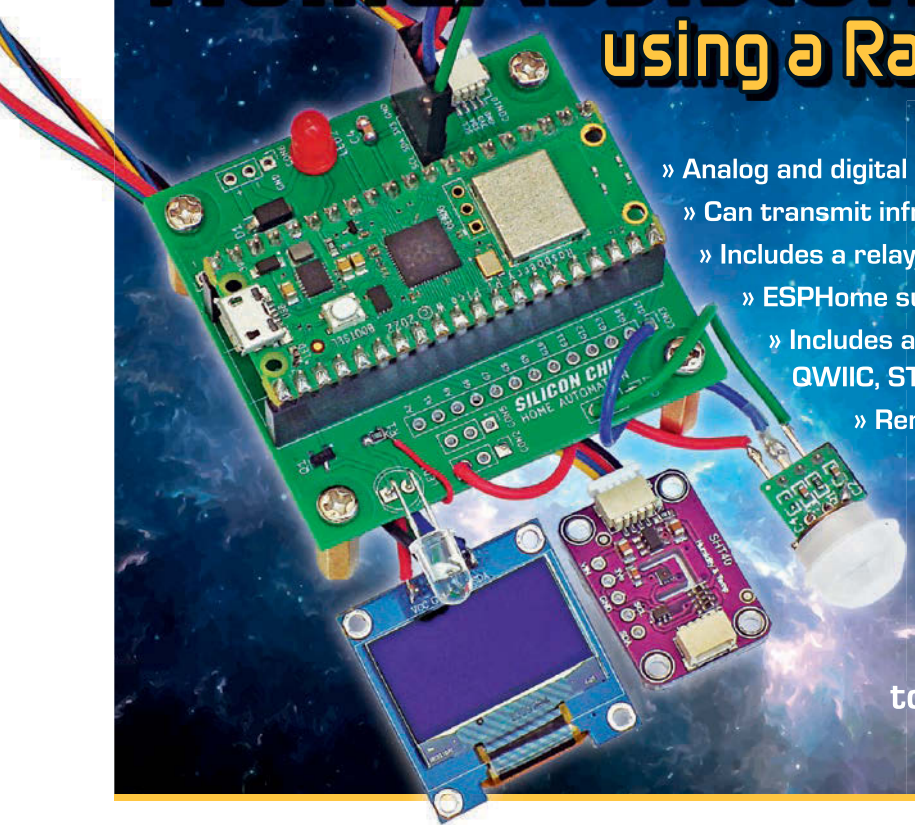
That article will describe how to use it to create a simple thermostat, demonstrating HomeAssistant's automation capabilities.

Next month, in a follow-up article, we'll add some more advanced features such as custom dashboards, remote access from a mobile phone or tablet, IR remote control, notifications and a camera or cameras. PE

Project by Richard Palmer

HomeAssistant Satellite

using a Raspberry Pi Pico W



- » Analog and digital inputs and outputs
- » Can transmit infrared remote control codes
- » Includes a relay/LED driver
- » ESPHome supports hundreds of sensors and controls
- » Includes an I²C bus connector compatible with QWIIC, STEMMA QT, GROVE & PiicoDev
- » Remote configuration and management

This simple board lets you connect lots of different kinds of sensors, displays and other things wirelessly to a HomeAssistant based Home Automation system.

In a Home Automation system, sensors and actuators attach to 'satellites', which can be placed at different locations around the home. Each can support multiple attached devices communicating with the hub over the household WiFi network.

Our Satellite uses a Pico W microcontroller module programmed with the ESPHome firmware. While the newer Pico 2 W may be used, it costs a little more and the extra power is not required for this application.

A small PCB hosts the Satellite's basic components and makes it easy to connect to a wide range of supported devices. You can see a list of the supported sensors at https://esphome.io/components

Circuit description

Fig.1 shows the Satellite circuit. As you can see, there isn't a lot to it besides the Raspberry Pi Pico W module shown in the middle.

Most sensors and many control devices will connect to the Satellite via a two-wire I²C serial bus. I2C0's SDA (I²C data) and SCL (I²C clock) functions are connected to GPIO pins 20 and 21, with pull-up resistors to the 3.3V rail as required for I²C communications.

Access to the I²C bus is provided by CON9, a 4-pin JST-SH connector that is compatible with QWIIC, Stemma QT, Grove and PiicoDev devices. CON8 also provides access to the I2C0 bus, allowing the connection of a small OLED screen.

If two devices with the same I²C address need to be connected, a second I²C bus is available at several locations on the expansion connector, CON4. However, note that pull-up resistors will need to be added if the second I2C1 bus is used.

CON4 expansion header pinout

Pin	Signals
1	GP15, I2C1, SPI1
2	GP14, I2C1, SPI1
3	GP13, SPI1, UART0
4	GP12, SPI1, UART0
5	GP11, I2C1, SPI1
6	GP10, I2C1, SPI1
7	GP9, SPI1, UART1
8	GP8, SPI1, UART1
9	GP7, I2C1, SPI0
10	GP6, I2C1, SPI0
11	GP28, ADC0
12	GP27, ADC1
13	GP26, ADC2

The adjacent table lists the functions available on the expansion connector. Separate pads for the 5V and 3.3V supplies and ground are provided at CON1-CON3.

For remote control, IR LED1 is driven by Q1, a BC817 or similar NPN transistor, from the 5V supply. Its 40mA operating current is higher than can comfortably be supplied directly by a digital output pin on the Pico module.

Transistor Q2 can drive an off-board 5V relay or solid-state relay (SSR). LED2 lights when the relay is operating, while diode D1 protects the transistor against back-EMF when a traditional relay switches off.

The 64kHz PWM signal at GP16 is smoothed by an RC low-pass filter formed by a 10µF capacitor and 470Ω resistor, giving a -3dB frequency of approximately

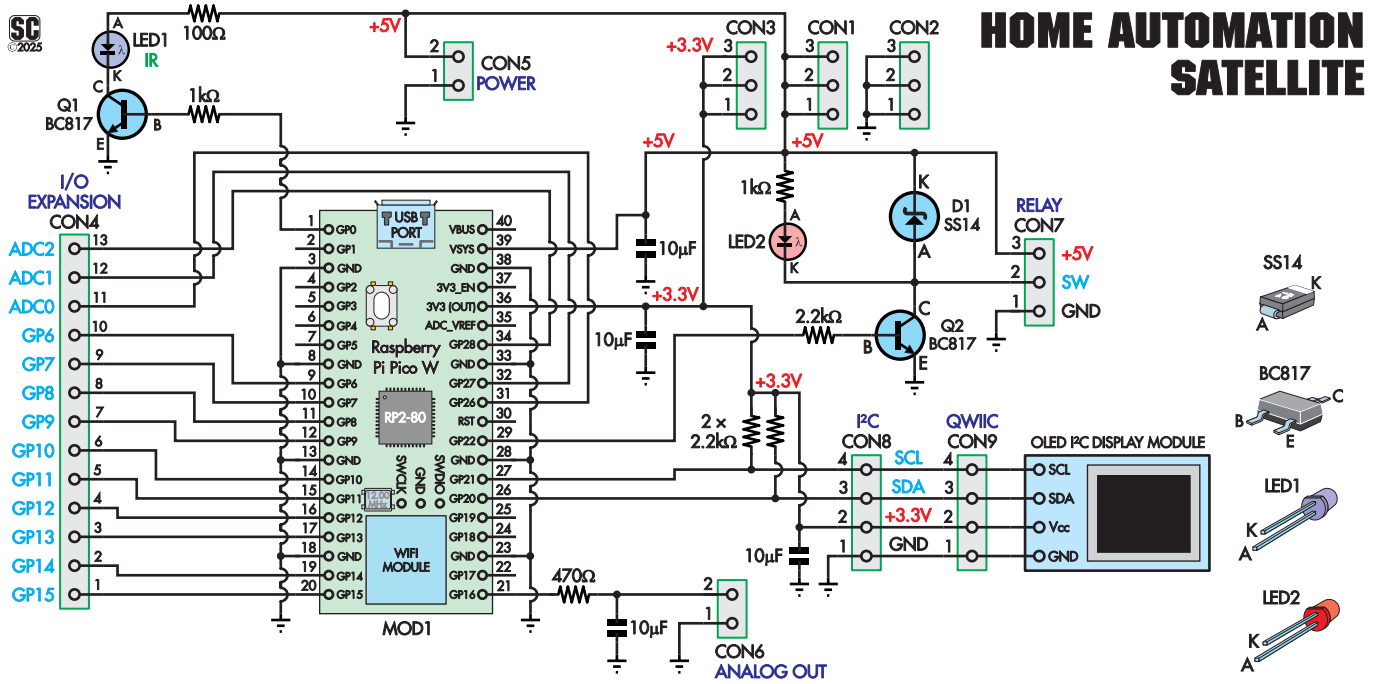


Fig.1: the Raspberry Pi Pico W module drives the IR LED (LED1) and relay through transistors Q1 and Q2. Digital and analog I/O is via CON4, with power available from CON1-3. CON9 is a QWIIC, Grove and PiicoDev compatible I²C connector. The I²C OLED display connects via CON8.

34Hz, reducing the ripple of a 50% duty cycle output to around 10mV peak-to-peak.

While the Pico is capable of much higher PWM frequencies, which would result in even lower ripple, there is a trade-off between frequency and bit depth (resolution), which is 10 bits at this frequency. The smoothed output is presented at CON6, forming a basic kind of digital-to-analog converter (DAC).

The board's 5V supply is powered either via a micro-USB cable plugged into the Pico W, or from CON5. In most cases, a 5V 500mA supply should be adequate, as the Pico W draws less than 100mA and the IR transmitter requires 20mA on average.

However, the supply must also be able to support the needs of any connected devices. For instance, 5V relays commonly draw 75mA or more when energised.

The Pico W module has an onboard 3.3V regulator capable of supplying 250mA to auxiliary devices. If more than this is required, an additional 3.3V supply will be needed.

Construction

The Satellite is built on a double-sided PCB coded 15104251 that measures 51 × 54mm. Start by soldering the SMD components in the locations shown in Fig.2. The band on the diode must point toward the power header.

Next, add the two 20-pin female headers for the Pico, the JST-ST connector (CON9) and the two LEDs. Leave CON1-CON8 off at this stage.

Solder headers to the Pico W, if they weren't already fitted, and plug it in with the USB socket at the top of the board. You can use the sockets on the board as a jig to hold the headers while soldering them to the Pico W.

Now plug the USB cable into a port on your PC. 5V should appear on CON1 and 3.3V at CON3. Disconnect it after checking those supply rails.

Configuration

If you have difficulty with any of the configuration steps below, there is documentation at ESPHome's website (https://esphome.io) and a helpful user community at https://community.home-assistant.io

Load the ESPHome firmware next. Go to the ESPHome Builder tab in HA, click on the + NEW DEVICE button and CONTINUE. You can name your device "myHome" (or something else if you prefer). Select Raspberry Pi Pico W as the platform, then click INSTALL. A copy of the encryption key will be saved in the configuration file.

Select the "Manual download" option when the pop-up window appears – see Screen 1. It will take a minute or more for HA to compile customised code for the new device. When the wheel stops spinning, select Download project and UF2 factory format, then close the pop-up window.

You may need to approve the download in your browser ("Keep" in Chrome) as it's coming from an unsecured host.

Hold down the BOOTSEL button on the Pico while plugging its USB cable into your computer. Copy the downloaded UF2 file across to the "RPI-RP2" drive that has appeared in the PC's file manager and close the pop-up window.

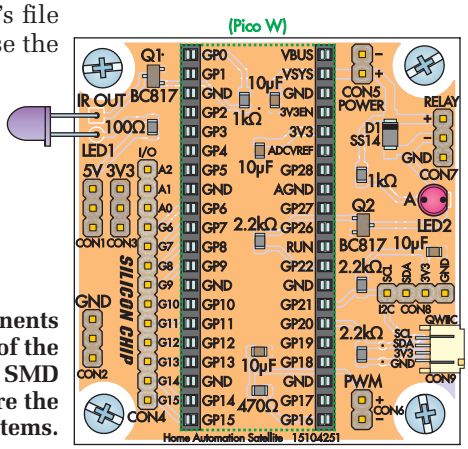


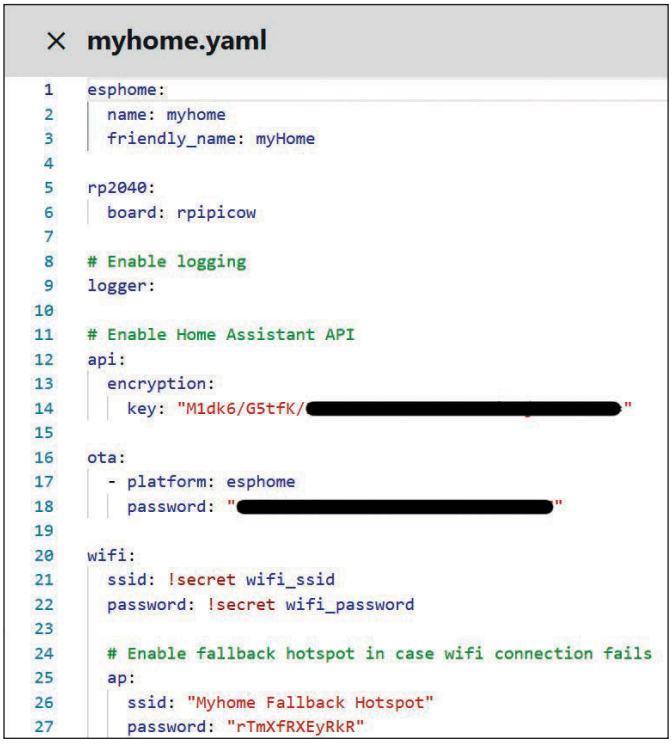
Fig.2: all components are on the top of the PCB; mount the SMD components before the through-hole items.

The block with your Pico's name should become ONLINE in the ESPHome Builder tab, after a 30 second delay (Screen 2). This is a basic ESPHome device, with no sensors or outputs configured. Click on the EDIT link on the myHome card and the configuration file will be displayed. Additional configuration lines for sensors and controls will be added below this code – see Screen 3.

Install ESPHome via the USB drive
You can install your ESPHome project myhome.yaml on your device via your file explorer by following these steps:
1. Disconnect your Raspberry Pi Pico from your computer
2. Hold the BOOTSEL button and connect the Pico to your computer. The Pico will show up as a USB drive named RPI-RP2
3. Download project preparing download...
4. Drag the downloaded file to the USB drive. The installation is complete when the drive disappears
5. Your Pico now runs your ESPHome project
BACK
CLOSE

Screen 1: the "New device" screen while the code is being compiled.

Screen 2: the initialised Pico W Satellite's card in the ESPHome builder tab.



Screen 3: the myHome device's initial configuration file.

Each YAML block starts with an identifier, usually the item's type. Values are indented and the number of leading spaces is significant (usually two per level). Each level's information is terminated when a less indented item is encountered. Comments begin with a "#" and continue to the end of line.

Close the editing window by clicking the X at the top left corner and click on the LOGS link in the myHome card. A window will open with the history of the device since it was last booted. In this case, most of the information is in purple, describing the WiFi connection process. Close the window.

The Satellite is now ready and the final step is to tell the HA Core that it exists.

Go to Settings then Devices & services and +ADD INTEGRATION. Search for "ESPHome" and click SUBMIT. The Satellite should pop up as a discovered device. If not, type "myHome.local" (or prefixed with whatever name you gave it) into the "Host" field and click SUBMIT.

If you had to enter the host name, the installer will ask for the encryption key from the myHome.yaml configuration file. Click SUBMIT after that.

The Area field may be left blank on the next screen. Approve the request for the device to perform HomeAssistant actions, then click FINISH to make the Satellite active. The Satellite may not appear on the Overview dashboard at this stage, as it has no sensors or controls defined.

Adding a sensor

Unplug the USB cable, connect an SHT40 temperature sensor to the I²C port and plug the USB cable back in. Reopen the editor and add the following code at the bottom of the myHome.yaml file.

```
## Temperature.yaml
## Temperature & Humidity sensor
## Pico W I2C0 bus
i2c:
  sda: 20
  scl: 21
  scan: true
  frequency: 400kHz

## SHT40 sensor
sensor:
  - platform: sht4x
    update_interval: 10s
    humidity:
      name: "Humidity"
      id: humidity
    temperature:
      name: "Temperature"
      id: temperature
```

Rather than typing this in, you can copy and paste this from the file named "Part2.yaml" in the download package for this series (siliconchip.au/Shop/6/2482). That will not only save time but ensure you get the correct formatting for it to work.

The first section defines the I²C bus that will be shared by other connected devices. The sht4x 'platform' uses the defined I²C bus by default. The "name:" field is used when displaying the entity, while the "id:" field is required if the entity is to be referenced by code.

In the SHT40 section, humidity and temperature sensors are defined, and the interval between readings is shortened from the default value of 60 seconds to 10.

After adding the code, click SAVE and INSTALL (in the top right-hand corner) and select the Wirelessly option in the pop up. Depending on your HA setup, compiling the code may take a minute or more. You will know it has finished when the green SUCCESS message appears in the log.

The code will then attempt to upload to the Pico and, after a delay of thirty or more seconds; the screen will show the Pico's boot log. At the bottom should be light blue text indicating that temperature and humidity readings are being sent to the HA hub.

If your configuration won't compile, it may be due to incorrect indentation or code in the wrong location.

The window may be closed by clicking STOP at any time after the download has completed. While STOP may appear a strange command to close the window, HA has launched a virtual Docker machine to do this work, and stopping the virtual machine when it is no longer needed closes the window.

Close the myHome.yaml edit window, then go to HA's Overview dashboard. After a minute or so, the myHome card should appear, showing the temperature and humidity. If it doesn't, try reloading the web page.

Controlling the relay and LED

Only a few extra configuration lines are needed to map the relay driver's GPIO pin as a switch output. They are added to the Satellite system's myHome.yaml configuration file below the SHT40 sensor definitions. The "switch:" component indicates that this is a binary output. The GPIO platform then maps the function to a microcontroller pin:

```
## Relay_LED.yaml
## Pico Relay and LED
switch:
  - platform: gpio
    pin: 22
    name: "Relay LED"
    id: led
```

Save and install the updated configuration. After a minute or so, the myHome tab on the Overview dashboard should include a switch for the Relay LED – see Screen 4. Toggling the switch should cause the LED to come on.

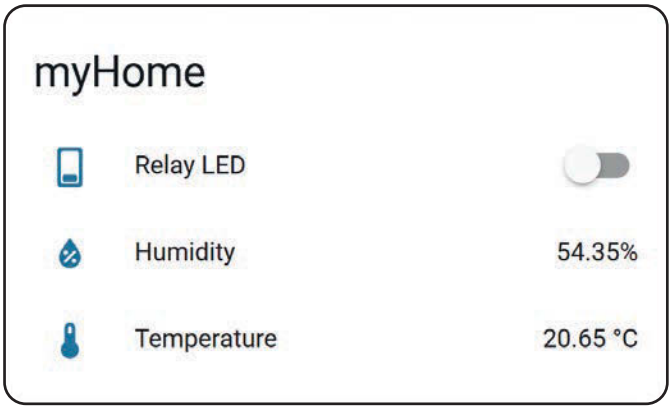
Adding a display

ESPHome supports a wide range of displays. The SSD1306 OLED used in this project has an I²C interface and we can use it to show the current temperature and humidity, along with the LED/relay state and a graph of temperature over time.

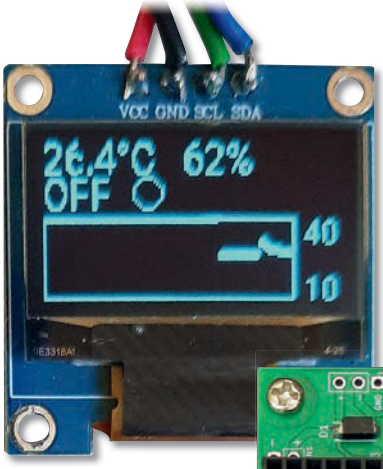
The display can be daisy-chained with the temperature sensor or connected via CON8. When wiring it up, take care with the pinouts as they tend to vary between display manufacturers.

The I²C bus is already defined, so all that is needed is to declare the OLED display and add some lines of code to run regularly that write to the display. The OLED screen is set to the standard I²C address for 128×64 pixel mode. As the built-in font is blocky, we can use some Google fonts instead.

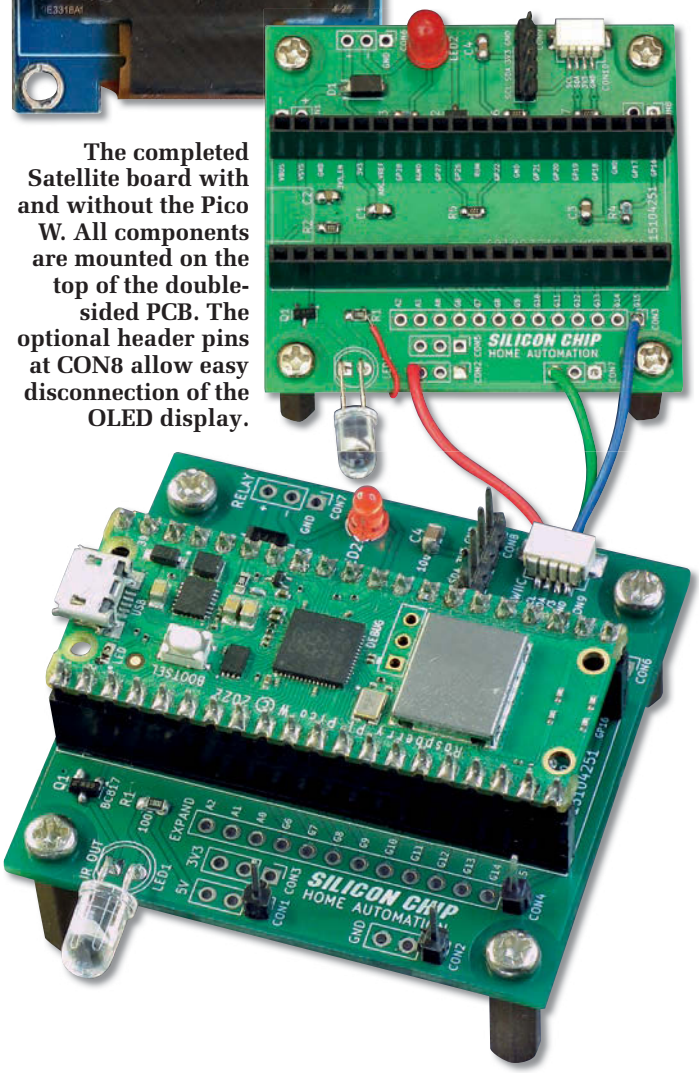
In the display section of the code below, there is a portion



Screen 4: the myHome dashboard card with the Relay LED.



The SSD1306 display showing temperature, humidity and the state of the Relay LED.



The completed Satellite board with and without the Pico W. All components are mounted on the top of the double-sided PCB. The optional header pins at CON8 allow easy disconnection of the OLED display.

Parts List – Home Assistant Satellite

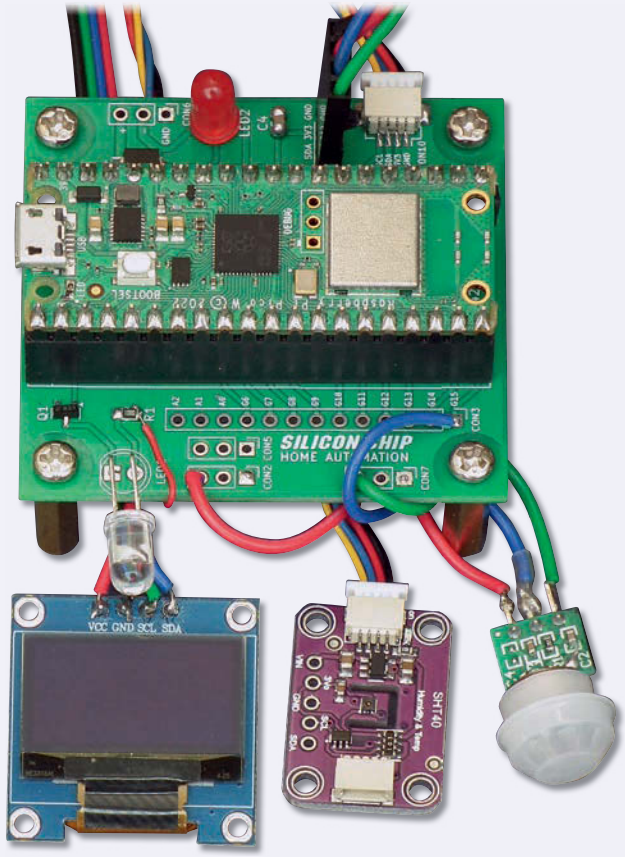
1 double-sided PCB coded 15104251, 51 × 54mm
1 Raspberry Pi Pico W module with two 20-pin headers (MOD1)
1 micro-B USB cable
2 20-pin single-row female header sockets
4 3-pin headers (CON1-CON3, CON7) (**optional**)
1 13-pin header (CON4) (**optional**)
2 2-pin headers (CON5, CON6) (**optional**)
1 4-pin header (CON8) (**optional**)
1 4-pin JST-SH 1mm-pitch socket (CON9)

Semiconductors
2 BC817 NPN transistors, SOT-23 package (Q1, Q2)
1 5mm 50mA IR LED (LED1)
1 3mm red, yellow, amber or green LED (LED2)
1 SS14 or similar 1A schottky diode, DO-214AC/SMA

Capacitors
4 10µF M2012/0805 16V X7R

Resistors *(all SMD M2012/0805 size 1%)*
3 2.2kΩ 2 1kΩ 1 470Ω 1 100Ω

Add-on sensors & display used in this article
1 SHT4x QWIIC temperature and humidity sensor
1 QWIIC cable
1 AM312/AS312 PIR sensor module
1 SSD1306 I²C OLED display



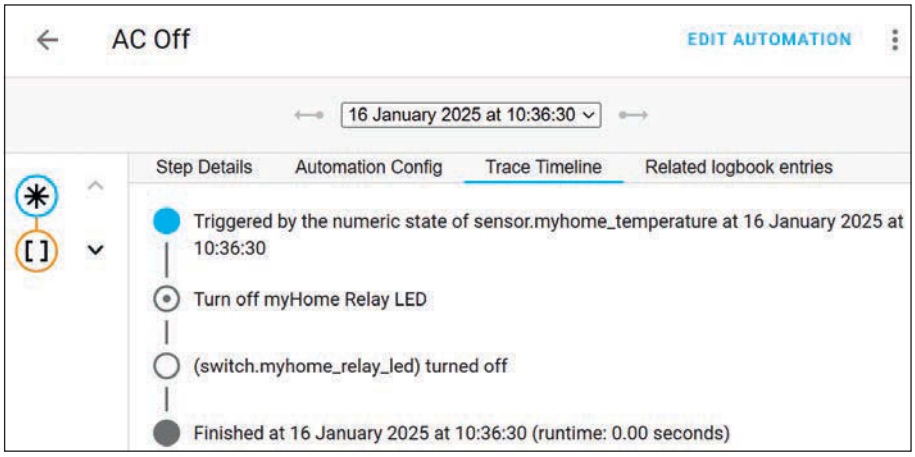
marked “lambda”. Any code following the lambda marker is passed to a command interpreter for execution at runtime. Lambda code is written in the C language.

Add the code to myHome.yaml below the relay’s “switch:” definition block:

```
## OLED_display.yaml
## define the fonts
font:
  - file: “gfonts://Roboto”
    id: roboto_12
    size: 12
  - file: “gfonts://Roboto”
    id: roboto_16
    size: 16

## graph configuration
graph:
  - id: temp_graph
    sensor: temperature
    ## sliding time window
    duration: 60min
    width: 106
    ## half display height
    height: 32
    border: True
    max_value: 40
    min_value: 10
    y_grid: 10

## SSD1306 display
display:
  - platform: ssd1306_i2c
    model: “SSD1306 128x64”
    address: 0x3C

## Print the text and graph
    lambda: |-
      it.printf(0, 0, id(roboto_16),
       “%.1f°C   %.0f%%”,
       id(temperature).state,
       id(humidity).state);
      if (id(led).state) {
        it.print(0, 14,
         id(roboto_16), “ON”);
        it.filled_circle(44, 22, 6);
      } else {
        it.print(0,14,
         id(roboto_16), “OFF”);
        it.circle(44, 22, 6);
      }
      it.print(110, 30,
       id(roboto_12), “40”);
      it.print(110, 52,
       id(roboto_12), “10”);
      it.graph(0, 32,
       id(temp_graph));
```

As with other YAML code, the lambda block ends when less indented text is found. The version in the download pack is easier to read, as the lines haven’t been wrapped to fit into a magazine column.

The lambda code prints temperature and humidity on the first line, with the state of the LED and a circular on/


AC Off
EDIT AUTOMATION
16 January 2025 at 10:36:30
Step Details
Automation Config
Trace Timeline
Related logbook entries
Triggered by the numeric state of sensor.myhome_temperature at 16 January 2025 at 10:36:30
Turn off myHome Relay LED
(switch.myhome_relay_led) turned off
Finished at 16 January 2025 at 10:36:30 (runtime: 0.00 seconds)

Screen 5: the Traces screen for the "AC off" automation.

off indicator on the second line. The bottom half of the screen displays a graph of the temperature over the past 60 minutes, between 10°C and 40°C, with horizontal grid lines every 10°C.

The state of the relay LED slider on the dashboard (Screen 4) should be reflected in the OFF/ON text and indicator on the display.

A simple thermostat

Automations use the output of a sensor or other input condition, such as the time of day, to trigger an action. We will create a simple air conditioner thermostat that switches on the relay/LED when the temperature rises above 28°C and off again when the temperature falls below 27°C. These settings are convenient for testing with a finger on the temperature sensor.

To implement the thermostat, the temperature reading is connected to the Relay/LED using two automations. On the HomeAssistant screen, go to Settings then Automations & scenes. Click on + CREATE AUTOMATION at the bottom right corner of the screen and Select "Create new automation" from the pop-up menu.

Under "When", click on + ADD TRIGGER. Select Device and choose "myHome" from the drop-down list. Select "myHome temperature changes" from the Trigger drop-down. Enter 28 into the Above field. The Below and Duration fields should be left blank.

Under "Then do", click ADD ACTION. Select Device and "myHome" as before. The Action is "Turn on myHome Relay LED". Save the automation as "AC On".

Click the back arrow at the top of the screen to return to the "Automations" menu and create a second automation to turn the AC off.

Under "When", click on + ADD TRIGGER. Select Device and then myHome from the drop-down list. Select "myHome temperature changes" from the Trigger drop-down. Enter "27" into the Below field. Under "Then do", click ADD ACTION. Select the Device myHome as before. The Action is "Turn off myHome Relay LED". Save the automation as "AC Off".

The two automations will automatically activate within a minute. Go back to the Overview dashboard and hold a finger to the temperature sensor. The LED should light once the temperature rises above 28°C. The indicator on the dashboard should change, the relay LED should light, and the OLED display should indicate that the virtual AC is on.

When the temperature is allowed to fall, the relay LED should switch off below 27°C.

Go back to the Automations & scenes menu and select "AC off". Click on TRACES at the top right corner. The Trace Timeline tab should produce a screen very much like Screen 5; this view is useful when debugging more complex automations.

We'll have a follow-up article next month that describes how to use IR remote control to switch a split-system air conditioner on and off, a more realistic control method.

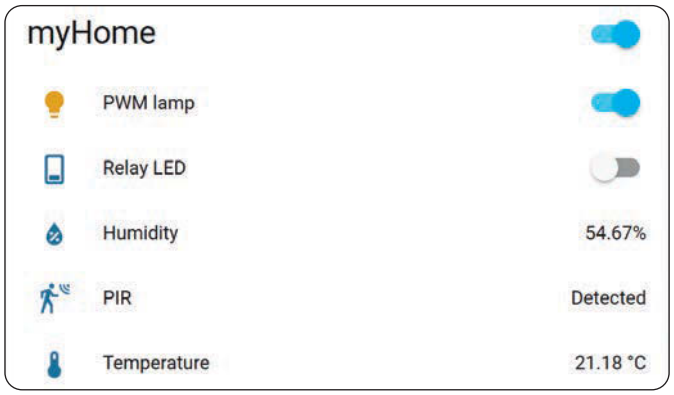
Analog outputs

We can set up GP16 to generate a 64kHz PWM signal that is smoothed by a simple RC filter and presented at CON6. The result is a voltage that's proportional to the temperature, with the output scaled to a maximum of 3.3V at 33°C.

The "monochromatic light" platform component is used as it provides a straightforward conversion from an input value to a brightness output. However, the conversion isn't linear by default; the gamma value must be explicitly set to 1.

Edit the configuration file's "sensor:" block to add the on_value clause immediately after the "id: temperature" statement, then add a new "output:" block for the PWM at the end of sensor clause:

```
## PWM_temp.yaml
## PWM set by temperature
## Replaces existing temperature
## and humidity sensor code
## sensor:
  - platform: sht4x
    update_interval: 10s
    humidity:
      name: "Humidity"
      id: humidity
    temperature:
      name: "Temperature"
      id: temperature
      on_value:
        then:
          - light.turn_on:
              id: pwm_lamp
              ## 10 degrees/volt
              brightness:
                !lambda "return x/33;"
```



Screen 6: the myHome dashboard card with the PIR sensor.

Also add the following after the sensor block:

```
## PWM_output.yaml
output:
  - platform: rp2040_pwm
    pin: 16
    id: PWM_16
    frequency: 64000Hz
light:
  - platform: monochromatic
    output: PWM_16
    ## straight line temp:pwm
    gamma_correct: 1.0
    name: "PWM lamp"
    id: pwm_lamp
```

When the code is installed, the PWM output voltage at CON6 will track the measured temperature.

PIR motion detection

If you need the motion detection feature, connect a PIR (passive infrared) motion sensor module to the GP15 input at pin 1 on CON4, with 5V power and ground to CON1 and CON2, respectively. The required connections are shown in Fig.3.

The sensor produces a binary value, so the code sits in a separate block to sensors that produce numeric values. Add the following code to your device in ESPHome Builder after the "sensor:" block, then save and install the changed configuration:

```
## PIR.yaml
binary_sensor:
  - platform: gpio
    pin: 15
    name: "PIR"
    device_class: motion
    id: PIR
```

The MyHome card on the Overview dashboard should now look something like Screen 6. PE

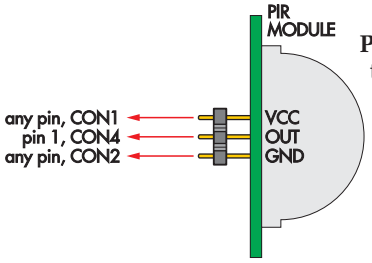
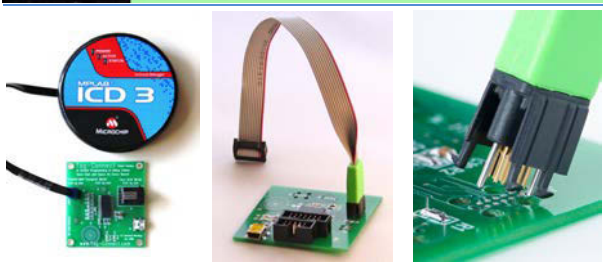


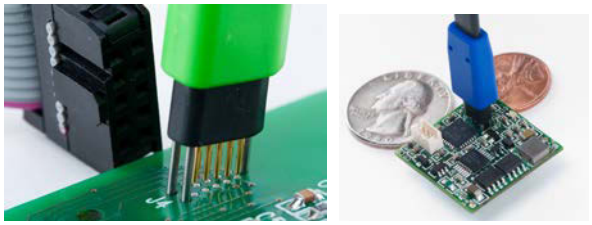
Fig.3: wiring for the PIR sensor is simple. In this example, V_{OUT} goes to GPIO GP15 (pin 1 on CON4), while 5V power comes from any combination of pins on CON1 & CON2.

Tag-Connect



JTAG Connector Plugs Directly into PCB!!

No Header! No Brainer!



Our patented range of Plug-of-Nails™ spring-pin cables plug directly into a tiny footprint of pads and locating holes in your PCB, eliminating the need for a mating header. Save Cost & Space on Every PCB!!

Solutions for: PIC . dsPIC . ARM . MSP430 . Atmel . Generic JTAG . Altera Xilinx . BDM . C2000 . SPY-BI-WIRE . SPI / IIC . Altium Mini-HDMI . & More

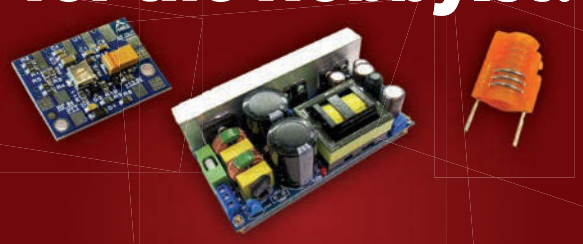
www.PlugOfNails.com

Tag-Connector footprints as small as 0.02 sq. inch (0.13 sq cm)

ENIGMA COMPONENT SHOP

www.enigma-shop.com

Electronic Components for the Hobbyist!



Cable & Wire • Capacitors • Connectors • Filters
Kits & Modules • Hardware & Cooling
and much more...

READY TO BE REWARDED? Join our loyalty scheme and earn points on every purchase you make which can be used for discounts in the future.

Read our reviews on Trustpilot

Max's Cool Beans

By Max the Magnificent

Weird & Wonderful Arduino Projects

Part 21: Ant automata antics and testing times

You may already realise this, but dilly-dallying is not the same as shilly-shallying. You see, dilly-dallying is wasting time by being slow, aimless, or dawdling (the art of taking forever to get nowhere), while shilly-shallying is specifically being indecisive or hesitant to make a choice or take any action ("Shall I or shall I not?"). Let's not do either – we'll get straight back into it!

Plaintive pleas

I've been receiving plaintive pleas from readers who've created one of our retro game consoles, essentially saying, "Don't forget about us!"

The gist of these messages is that, although they are captivated by our current Chronological Contraption project and while they are certainly looking forward to future Arduino-based adventures, they would also like us to keep suggesting new and interesting things to do with their consoles. Who amongst our number could fault them for that?

As a reminder, the console features a 14×10 array of tricolour light-emitting diodes (LEDs). I have a large 'to-do list' of games I intend to implement on the console in the fullness of time.

These include puzzle and strategy games like Mastermind, Lights Out and Minesweeper, plus arcade-style games like Breakout, Pong, Space Invaders, Frogger and Maxtris (a fun game that's legally distinct from Tetris).

Additional contenders are board and logic games, such as Tic-Tac-Toe (not exciting on its own, perhaps, but useful as a demonstration project), Connect Four and Reversi (Othello). Cellular automata and simulations are also perennial favourites. Speaking of which…

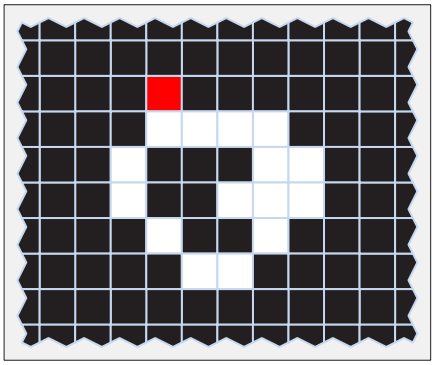


Fig.1: a trail left by Langton's ant.

Ant automation antics

Just saying "cellular automata" has unleashed the beast within me. I think we should use our consoles to implement classics like Conway's Game of Life, Seeds and Brian's Brain. I don't pretend to be a clairvoyant, but I'm sure I can see all of these in our future. For the moment, however, let's kick off with Langton's ant, which was invented by Chris Langton in 1986.

Langton's ant isn't a cellular automaton in the strict sense. Rather, it's a two-dimensional Turing machine with a very simple set of rules but complex emergent behaviour.

We start with all the cells in our array switched off (black). After that, we arbitrarily (randomly) select one of the cells to be our 'virtual ant' and assign it a random direction (north, south, east or west). Let's colour our ant red so we have a clue where it is and what it's doing.

The rules are as simple as can be. If the ant finds itself on a white cell, it flips the colour to black, turns 90° clockwise, and moves forward one square. Conversely, if the ant finds itself on a black cell, it flips the colour to white, turns 90° anticlockwise, and again moves forward one square (see Fig.1).

Ideally, this would be implemented on an infinite array of cells. In our case, we're limited to our 14×10 'universe'. As we've done in the past, we will wrap things around, treating the left and right edges of the array as though they touch, so that if the ant wanders off one side, it immediately reappears on the other.

We will do the same for the top and bottom edges, effectively turning our rectangular universe into a doughnut-shaped surface that mathematicians call a topological torus.

Cunning code

We'll use the version of the snake program that we created in the June 2026 issue of *PE* as our starting point. You can find a copy in the file named **CB-sep26-code-01.txt**. As usual, all files mentioned in this column are available as part of the September 2026 download package from the PE website at https://pemag.au/link/accl

Following the path of least resistance, we'll keep most of this code as-is, even if we aren't using it, because you never know if it will come in handy in the future. The new version reflecting the edits we're about to discuss can be found in the file named **CB-sep26-code-02.txt**.

The first big changes occur in the "Game-Related Stuff" section. We replace the definitions and declarations from our old snake program with their equivalents required by our new ant program. The updated versions are shown in Listing 2(a). We're using the convention that the listing number (2 in this case) corresponds to the numerical part of the matching code file name ("02" here).

Most of this should be reasonably understandable. One point worth noting is that we've modified our original direction definitions, which previously

```
#define TICK                1
#define ANT_MOVE_TICKS    500
#define COUNT_DOWN_DELAY  500

#define NUM_DIRECTIONS      4
#define NORTH               0
#define EAST                1
#define SOUTH               2
#define WEST                3

#define ANT_COLOR         RED

uint8_t  AntDirection;
uint8_t  AntX;
uint8_t  AntY;
uint32_t CellUnderAnt = BLACK;

uint32_t TimeThen = millis();
uint32_t TimeNow;
uint32_t AntThen = TimeThen;
```

Listing 2(a): ant-specific definitions and declarations.

were NORTH = 0, SOUTH = 1, EAST = 2 and WEST = 3. Our new versions (lines 145-148) are NORTH = 0, EAST = 1, SOUTH = 2 and WEST = 3.

The reason for this new scheme is that incrementing our AntDirection variable is now equivalent to a 90° clockwise rotation, while decrementing it is equivalent to a 90° anticlockwise rotation. There's a subtlety here regarding wrapping around, which we'll get to shortly.

Observe the CellUnderAnt variable on line 155. We're going to use this to track the original colour of the cell that's currently sitting under the ant (remember we're displaying the ant itself in red).

Also, we've modified the setup() function so that instead of calling the old InitializeSnake() function, it now calls a new InitializeAnt() function. Similarly, we've modified the loop()function so that instead of calling the old UpdateSnake() function, it now calls a new UpdateAnt() function.

We'll leave all the fundamental switch-handling code as-is, simply removing any snake-related actions since we won't be using them.

Initialising the ant is easy-peasy lemon squeezy, as seen in Listing 2(b). After clearing the array to black on line 465, we pick random X and Y values for the ant's initial location on lines 468 and 469, respectively. Observe that we lightly constrain this location to the central area of the array. On line 470, we pick a random direction for our ant to commence its travels.

In an earlier column, we created two handy-dandy functions called PeekPixel() and PokePixel(). These don't directly access the physical NeoPixels on the console. Instead, they read and write the colours stored in the Arduino's in-memory representation of the display, which we can visualise as a virtual array.

```
void InitializeAnt ()
{
  // Clear the 14 x 10 array
  DrawRectSolid(0, 0, MAX_X, MAX_Y, BLACK);

  // Generate random location and direction
  AntX = random(5, 10);
  AntY = random(4,  6);
  AntDirection = random(0, NUM_DIRECTIONS);

  // Draw the ant on the array
  CellUnderAnt = PeekPixel(AntX, AntY);
  PokePixel(AntX, AntY, ANT_COLOR);
  Neos.show();
}
```

Listing 2(b): initialising the ant.

```
void UpdateAnt ()
{
  int8_t   deltaX;
  int8_t   deltaY;
  int8_t   deltaD; // Direction

  if ( (TimeNow - AntThen) >= ANT_MOVE_TICKS) {
    // Update direction and old cell color
    if (CellUnderAnt == BLACK) {
      deltaD = -1; // 90 degrees anticlockwise
      CellUnderAnt = WHITE;
    }
    else // CellUnderAnt ==  WHITE
    {
      deltaD = 1;  // 90 degrees clockwise
      CellUnderAnt = BLACK;
    }

    PokePixel(AntX, AntY, CellUnderAnt);
    AntDirection = ((AntDirection + NUM_DIRECTIONS) + deltaD)
                                                  % NUM_DIRECTIONS;

    // Generate next X-Y location for the ant
    if (AntDirection == NORTH)
      {deltaX =  0; deltaY =  1;}
    else if (AntDirection == SOUTH)
      {deltaX =  0; deltaY = -1;}
    else if (AntDirection == EAST)
      {deltaX =  1; deltaY =  0;}
    else if (AntDirection == WEST)
      {deltaX = -1; deltaY =  0;}

    AntX = (AntX + NUM_COLS + deltaX) % NUM_COLS;
    AntY = (AntY + NUM_ROWS + deltaY) % NUM_ROWS;

    // Save color under ant and draw ant
    CellUnderAnt = PeekPixel(AntX, AntY);
    PokePixel(AntX, AntY, ANT_COLOR);

    Neos.show();
    PlayClick();

    AntThen = millis();
  }
}
```

Listing 2(c): updating the ant.

The LEDs themselves aren't updated until we call Neos.show(), which copies the values from the virtual array and streams them to the physical array. We're poised to use these little scamps now.

On line 473, we read the current colour of the cell (pixel) where we're going to place our ant and store a copy in our CellUnderAnt variable. On line 474, we load this cell with the colour of our ant, which we previously set to be red.

Finally, on line 475, we call the Neos.show() function, which will display our initial setup on the array. Not surprisingly, this means all the cells in the array will be black, apart from the randomly located ant, which will be red.

All that remains is our UpdateAnt() function, as illustrated in Listing 2(c). It's important to remember that whenever we enter this function, the ant is still pointing (heading) in its previous direction. Our task is to determine the ant's new direction, move the ant to its new location, and tidy up after ourselves by setting the ant's original cell to its updated colour (state).

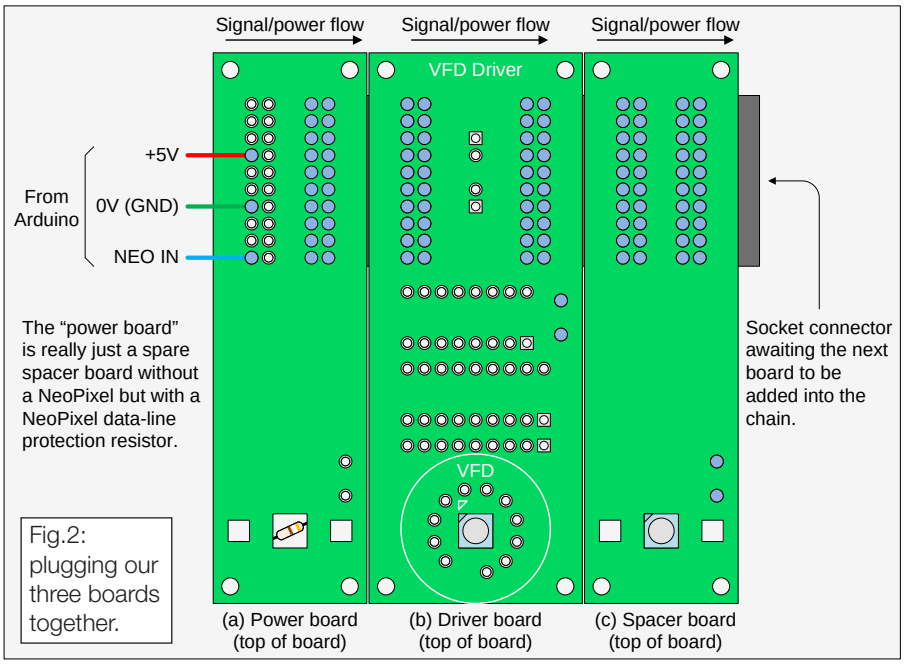

Signal/power flow
Signal/power flow
Signal/power flow
VFD Driver
From Arduino
+5V
0V (GND)
NEO IN
The "power board" is really just a spare spacer board without a NeoPixel but with a NeoPixel data-line protection resistor.
Socket connector awaiting the next board to be added into the chain.
VFD
(a) Power board (top of board)
(b) Driver board (top of board)
(c) Spacer board (top of board)

Fig.2: plugging our three boards together.

This is really rather simple. Previously, we defined ANT_MOVE_TICKS as 500ms (milliseconds), or half a second, which means our ant will scamper across the array twice each second. We can speed things up or slow them down by decreasing or increasing this value, respectively.

On lines 480-482, we declare some 8-bit integer variables to hold any delta (change) values in the ant's X-Y location and its direction.

On line 484, we check whether it's time to move the ant. Assuming the time has come, on line 486, we check if the colour of the cell under the ant was originally BLACK. If so, we set deltaD to -1 (to rotate the ant 90° anticlockwise) and we update the colour of the cell to WHITE. Otherwise, we set deltaD to +1 (to rotate the ant 90° clockwise) and we update the colour of the cell to BLACK.

On line 496, while we're thinking about it, we use our PokePixel() function to load the new colour of the cell into the virtual array. On line 497, we generate our new direction. Although we've seen this sort of thing many times before, it's worth reminding ourselves how this statement works because we'll be seeing a lot of it in the future.

Our AntDirection variable can take one of four values: 0, 1, 2 and 3, corresponding to NORTH, EAST, SOUTH and WEST, respectively. We can think of this as being a circular list: NORTH → EAST → SOUTH → WEST → NORTH…

Turning right means moving one step clockwise around this circle, while turning left means moving one step anticlockwise. Since the directions wrap around, turning right from WEST (3) takes us back to NORTH (0), and turning left from NORTH (0) takes us to WEST (3).

The statement on line 497 performs this wrap-around automatically. We've already established that the deltaD value is either +1 (turn right) or -1 (turn left). We first add NUM_DIRECTIONS (4) to ensure that our intermediate result can never be negative. This doesn't affect the final answer because adding any multiple of four will leave the result unchanged once we apply the modulo operator.

We then add deltaD, to apply the change in direction. Finally, we use the modulo operator (%) to constrain the result to the range 0 to 3.

For example, if the ant is facing WEST (3) and turns right, the calculation becomes (3 + 4 + 1)%4 = 8%4 = 0, so the ant ends up facing NORTH. Similarly, if the ant is facing NORTH (0) and turns left, the calculation becomes (0 + 4 – 1)%4 = 3%4 = 3, so the ant ends up facing WEST. Yes, you're right, a little "Tra-la" is certainly in order.

Now that we know the direction in which our ant is headed, on lines 500-507, we generate the corresponding deltaX and deltaY values, each of which can be -1, 0, or +1.

On lines 509 and 510, we calculate the AntX and AntY values associated with the ant's new location. Once again, we use the modulo operator to make the ant wrap around the edges of the array. Thus, if the ant wanders off one side, it reappears on the opposite side.

On line 513, we read the current colour of the cell (pixel) where we're going to place our ant, and we store a copy of this colour in our CellUnderAnt variable. On line 514, we load this cell with the colour of our ant, which we previously set to be red.

On line 516, we call the Neos.show() function, which will display the current state of play on the array.

On line 517, we play the click sound effect that we originally created for our snake. This means we'll hear a click every time the ant moves.

Last, but certainly not least, on line 519, we update our AntThen variable to the current value returned by millis(). This resets our timing reference, so the next move won't occur until another ANT_MOVE_TICKS milliseconds have elapsed.

Now it's time to upload our code into the console and watch this first incarnation of our Langton's ant as it scampers around the array performing its curious little ant dance. At first glance, its movements appear almost random, but appearances can be deceiving…

The chronological contraption

And so, having slaked the appetites of the retro games console brigade (at least for the moment), we return to our Chronological Contraption.

As you'll no doubt recall, by the end of our previous column, we'd added a NeoPixel and a ceramic capacitor to one of our driver boards, and similarly, to one of our spacer boards. We also added a plug connector and a socket connector to each board. Both are right-angle connectors with 20 pins arranged in two rows of 10.

All of these components are inserted from the bottom side of the board and soldered on the top side. The only component that will, in the fullness of time, be added to the top side of each driver board is the vacuum fluorescent display (VFD) tube. Also, for the spacer boards, we'll eventually add any hardware required to display separator characters, such as colons in the **hh:mm:ss** time field.

We also used a spare spacer board to implement a simple power board. Since all we're going to do at the moment is test our NeoPixels, we connected only three flying leads to this board: +5V, 0V (GND) and a signal to drive our chain of NeoPixels. The other ends of these leads have male pins to plug into our Arduino's headers.

Also, we added a 390Ω ¼W resistor (with orange [3], white [9], and brown [×10] coloured bands) to the power board between the data in and out pins of the unused NeoPixel cutout (anything between 330Ω and 470Ω should do the trick). We're using this resistor to tame the fast edges from the Arduino's output, reducing ringing and any associated overshoot and undershoot on the data line.

Our penultimate task was to plug all three boards together, as illustrated

in Fig.2. Then we connected all three flying leads to our Arduino Uno. We decided to use digital pin 6 to drive the NeoPixel chain for no better reason than that this happens to be my lucky number.

I left you with a mission to write a small Arduino program to exercise our NeoPixel chain and verify that everything is working as expected before we assemble the remaining boards. How well did you do?

One, two, three, lots!

My wife (Gigi the Gorgeous) and I teach a class of twenty or so three-to-five-year-olds at Sunday School. I'm in charge of the puppet show with Chickpea the chicken (who says she's three) and Biscuit the dog (who claims to be four). The kids love it when the puppets try to count things, saying, "One, two, three… lots!"

The reason I mention this here is that, as we previously discussed, the way we've architected things with individual driver and spacer boards means you can implement your system using as few or as many as you wish. At one end of the spectrum, you could display all the digits one after another on a single tube.

Alternatively, you could opt for a "Full Monty" implementation like mine. This will employ 14 driver boards and 5 spacer boards (**YYYY:MM:DD hh:mm:ss**), for a total of 19 boards, each with its own NeoPixel. I'm going to assume this full-up implementation in our future discussions, which leads us to…

Testing, testing

My first humble program offering is shown in Listing 3(a). We've seen all of this before in one form or another. On line 1, we include the NeoPixel library. On line 3, we set the number of boards (the ones containing NeoPixels) in our chain to 2 (we can increase this later as we add more boards). On line 4, we specify the pin we're going to use to drive the NeoPixel string.

Since we intend to visually inspect each NeoPixel, we've set the on and off times to one second and 1/10th of a second on lines 5 and 6, respectively. This will give our eyes and brains sufficient time to register what's going on. We can speed things up later once we are happy that everything is working as planned.

On lines 8-12, we define a small collection of colours. For the purposes of these initial experiments, we're setting things to only 25% of full power because NeoPixels are extraordinarily bright and we don't want to hurt our eyes.

```
1  #include <Adafruit_NeoPixel.h>
2
3  #define NUM_BOARDS      2 // Number of driver/spacer boards
4  #define PIN_NEOS        6 // Pin used to drive NeoPixels
5  #define ON_TIME      1000 // Time each pixel is on
6  #define OFF_TIME      100 // Gap between pixels
7
8  #define BLACK   0x000000U // We're only running
9  #define RED     0x400000U // the NeoPixels at
10 #define GREEN   0x004000U // 25% of full brightness
11 #define BLUE    0x000040U
12 #define WHITE   0x404040U
13
14 #define NUM_COLORS       4 // Number of test colors
15
16 const uint32_t ColorWheel[NUM_COLORS] =
17 {
18   RED, GREEN, BLUE, WHITE
19 };
20
21 // Instantiate the NeoPixel string
22 Adafruit_NeoPixel Neos(NUM_BOARDS, PIN_NEOS, NEO_GRB + NEO_KHZ800);
23
24 void setup ()
25 {
26   Neos.begin();
27   Neos.clear();
28   Neos.show();
29 }
30
31 void loop ()
32 {
33   // Cycle through colors
34   for (int iCol = 0; iCol < NUM_COLORS; iCol++)
35   {
36     // Cycle through boards
37     for (int iBrd = 0; iBrd < NUM_BOARDS; iBrd++)
38     {
39       Neos.setPixelColor(iBrd, ColorWheel[iCol]);
40       Neos.show();
41       delay(ON_TIME);
42       Neos.setPixelColor(iBrd, BLACK);
43       Neos.show();
44       delay(OFF_TIME);
45     }
46   }
47 }
```

Listing 3(a): our two-board test program.

On lines 16-19, we declare an array of the colour values we've defined, excluding black, which we'll use to switch things off. This will make it easy for us to cycle through all the available options under program control.

The main area of interest here is our loop() function. Since we're running this directly from the Arduino, I decided to use a 'walking pixel' effect in which only one NeoPixel is active at a time.

On line 34, we declare an 'outer loop' to cycle through the colours. On line 37, we declare an 'inner loop' to cycle through the boards.

The main reason for using a walking pixel effect rather than simply switching on all the pixels simultaneously is power consumption. Theoretically, when running at full power, a NeoPixel's red, green, and blue LEDs each consume 20mA (milliamperes), which means 60mA per device when displaying white at full (100%) power.

In reality, I've never seen NeoPixels draw more than 15mA per element (45mA per device), but it's best to assume a worst-case scenario so we never give ourselves cause to regret the error of our ways.

In the not-so-distant future, we're going to have an assembly boasting 19 boards. If we were to run all these pixels at full power simultaneously, we'd potentially be sucking 19 × 60mA = 1140mA (1.14A), which is way more than our little Arduino can supply.

By using a walking pixel, we will only ever need a maximum of 60mA. In fact, since we're running our pixels at only 25% of full power, we'll only be drawing a maximum of 15mA, which is barely worth mentioning, so I won't.

I just ran this program on my two-board assembly (I'm not counting the power board), and everything worked as planned. I never doubted myself for a moment (phew!).

More boards!

Since we currently have only two

boards with NeoPixels, our program simply appears to make them switch back and forth.

The next step is to add the NeoPixels, ceramic capacitors, and connectors to the remaining 13 driver boards and four spacer boards. In my case, I decided to populate an extra copy of each board type ("Just in cases", as Aurélia says in *Love Actually*).

It goes without saying, but I'll say it anyway: **use your anti-static mat and wrist strap**! In my case, my eyes aren't what they used to be, so I also use magnifying glasses.

I like types that offer multiple interchangeable magnification options, such as those from YOCTOSUN (**www.amazon.co.uk/dp/B0CBBVY3MQ**) – cheaper versions are available. That way, you can select the magnification level that works best for you.

These also provide eye protection, which is important because the last thing any of us needs is blobs of hot solder or boiling flux in our eyes.

Once you've populated all the boards, you need to add them to the chain and test them. You can do this one at a time if you wish, incrementing the NUM_BOARDS definition in the program each time. This would certainly be the safest approach (in case there's a bad connection, like a short circuit, somewhere) but it would also be slow and tedious.

Alternatively, if you're feeling bold, you could simply add all the boards and test them in one fell swoop. This is potentially much faster… just so long as nothing goes wrong.

I opted for a middle course. First, I removed the initial spacer board, added three more driver boards, and modified the program to test these four **YYYY** boards. Then I added the **MM** and **DD** driver boards, along with two spacer boards, and tested this new configuration. Once that was working, I continued adding boards in stages until the entire chain was assembled and verified.

Once I was confident that all of my NeoPixels were working, I experimented with different effects, such as reducing the ON_TIME and OFF_TIME values to 100 and 20, respectively. Later, I sent mini strings of two or three pixels racing down the chain together, but I'll leave you to work out how to do that for yourself.

To-may-to, to-mah-to

Before we plunge deeper into the fray with gusto, abandon and aplomb, it may make our lives easier if we define some terminology to ensure that we're all tap-dancing to the same skirl of the bagpipes.

There are two main ways of attaching components to printed circuit boards (PCBs).

One early approach that is still in use today is called through-hole technology (THT). It's not uncommon to see this spelt as "thru-hole" (naughty engineers). In manufacturing and assembly contexts, it's common to refer to such devices as leaded through-hole or lead through-hole (LTH) components.

Historically, for integrated circuits (ICs) intended for through-hole mounting, it was common for their pins to be arranged in two rows on either side of the package. Thus, these were originally known as dual in-line (DIL) packages, especially in the UK. Today, however, the term dual in-line package (DIP) has become the international standard used by manufacturers and data sheets.

Similarly, some components, such as resistor packs, are available with a single row of pins presented in what was traditionally known as a single in-line (SIL) package. In modern data sheets, these are more commonly referred to as single in-line packages (SIPs).

A more recent approach, surface-mount technology (SMT), involves attaching devices to the surface of the board. These surface-mount devices (SMDs) are smaller and lighter than their through-hole cousins, and their pins are much closer together, which often makes them more challenging for beginners to solder.

My co-conspirator on the Chronological Contraption is my chum Joe Farr. From the beginning, we determined to use through-hole components wherever possible to make it easy for readers to replicate our project. We also felt that, for a project with such a delightfully retro flavour, traditional through-hole parts were entirely in keeping with the spirit of the undertaking.

Dimples and notches

DIPs come in a variety of sizes, including 8-, 14-, 16-, 18-, 20-, 28-, and 40-pin offerings. As shown in Fig.3, some packages have a semicircular notch at one end, while others have a small circular dimple at one corner (some have both, just to make sure).

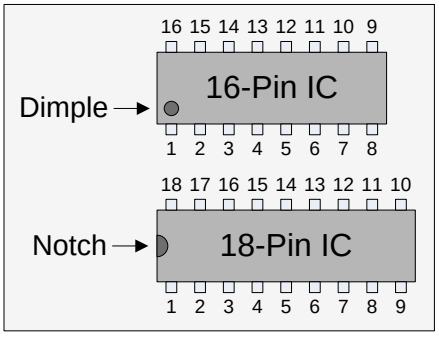


Fig.3: dimples or notches indicate pin 1.

A notch indicates which end contains pin 1, while a dimple marks the pin 1 corner. Once we've located pin 1, the remaining pins are numbered counterclockwise around the package when viewed from the top.

The reason I'm making a point of this is that, as we'll see next month, the pins on VFD tubes are numbered clockwise when viewed from the bottom of the tube.

Although these conventions sound different, they're actually the same thing viewed from opposite sides of the device.

When a board contains multiple DIP ICs, designers generally orientate them so that all the notches (or pin 1 indicators) point in the same direction (if they can!). This makes assembly, inspection, troubleshooting and replacement much easier.

On many older boards, especially computer and industrial designs, the pin 1 ends of the ICs were often facing toward the left-hand side or the top of the board. This wasn't a strict rule, but it was common enough that experienced engineers came to expect it.

Having said this, considerations such as signal flow and the need to minimise routing congestion may trump the desire for consistency. In such cases, one or more ICs may be rotated by 90°, 180° or 270° to simplify the PCB layout and reduce the number of tracks and/or vias required (where vias are plated holes used to connect tracks on different layers of the board).

If the designer is a tad unhinged, they may rotate parts in 45° increments, but let's ignore such obvious nonsense for now!

More components!

The next step is to separate the boards (dismantle the chain) and then add any remaining components to the driver boards. In our case, in addition to the components we've already attached, each driver board has a 16-pin DIP IC, an 18-pin DIP IC, a 9-pin SIP resistor pack and two electrolytic capacitors.

Remember that all of these components will be inserted into the bottom of the board, so anyone looking at the final Chronological Contraption will see only the tubes mounted on the top.

Before we add our new components, take a look at the bottom of

one of the driver boards, as illustrated in Fig.4(a). Observe that the silkscreen graphics for IC1 and IC2 include notches indicating the pin 1 ends. Also note that pin 1 itself is identified by a square pad (all the other IC pin pads are round).

You could solder your ICs directly to the board, but I strongly recommend against this. Although they add to the project's cost, IC sockets are worth their weight in gold if something goes wrong, such as inserting a chip the wrong way round (not that we would ever do anything so silly, of course). They also make it much easier to replace faulty ICs in the future should the occasion arise.

Now observe that the IC sockets have notches at one end. Make sure to align these notches with the ones on the silkscreen graphics. Once the sockets are soldered in place, they will conceal the graphics, so the socket notches will serve as our visual reference when we install the ICs later.

This is probably as good a time as ever to say that when it comes to holding the IC sockets and other components in place while you are soldering, painter's masking tape is your friend. It holds everything securely while you flip the board over, and it peels away cleanly afterwards without leaving any sticky residue behind.

Next, solder the 9-pin SIP resistor pack (labelled R1 on the silkscreen). There are a variety of different configurations available for this sort of package. The one we're using contains eight 10kΩ resistors. It's officially referred to as a 10kΩ bussed resistor network.

The word "bussed" (or occasionally "commoned") tells us that one end of all eight resistors is connected internally, sharing a common pin.

On the resistor pack illustrated in Fig.4, the common connection is pin 1, which will be identified by a dot, notch, bevel, or similar feature on the package. Other markings on the device may include "103", indicating 10 × 10^3 = 10,000Ω. Make sure that you insert pin 1 on the SIP into the correct pin 1 location on the board. As with the ICs, pin 1 is identified by a square pad on the board.

Last but certainly not least, add the two 63V 1µF radial-leaded electrolytic capacitors (labelled C1 and C2 on the silkscreen). Observe that the graphics on the silkscreen for these capacitors show both round and square pads. Also, the square pads have an associated '-' symbol indicating negative (-ve).

This is because, unlike our earlier ceramic capacitors, which can be inserted either way round, these electrolytic capacitors are polarised, which means they must be connected with the correct polarity. Inserting them the wrong way round can damage the capacitors (and more besides) and may prevent the circuit from working correctly.

The negative side of the capacitor, which is officially called the cathode (k), is usually indicated by a noticeably shorter lead. However, the most reliable guide is the stripe containing minus signs printed on the body adjacent to the negative lead.

Once you've soldered all the new components onto the driver boards, it

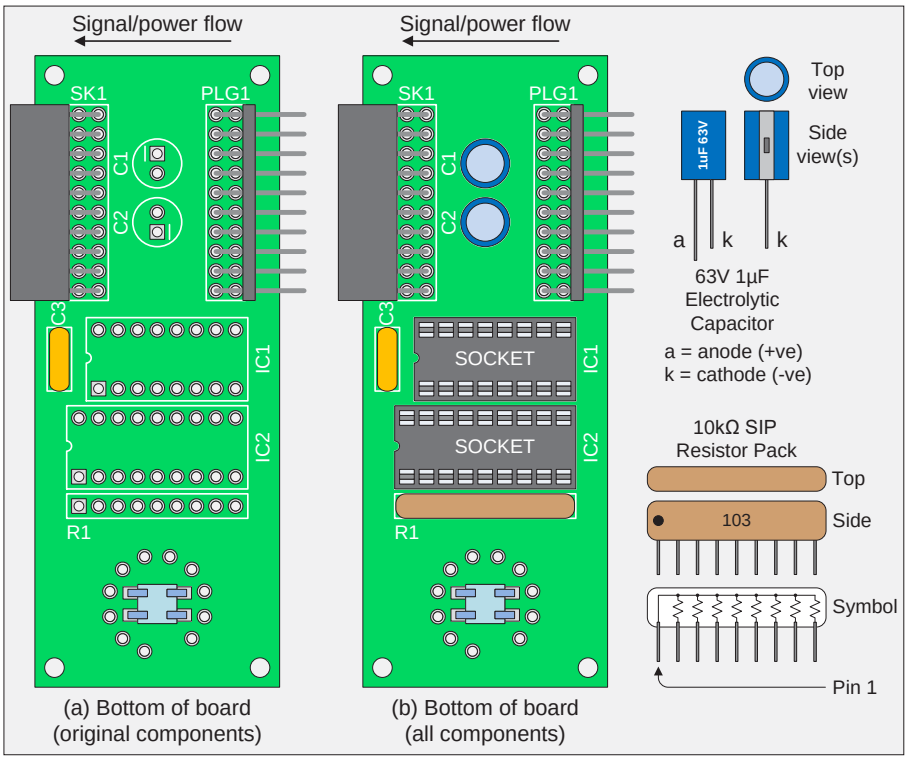


Fig.4: adding the remaining components to the driver boards.

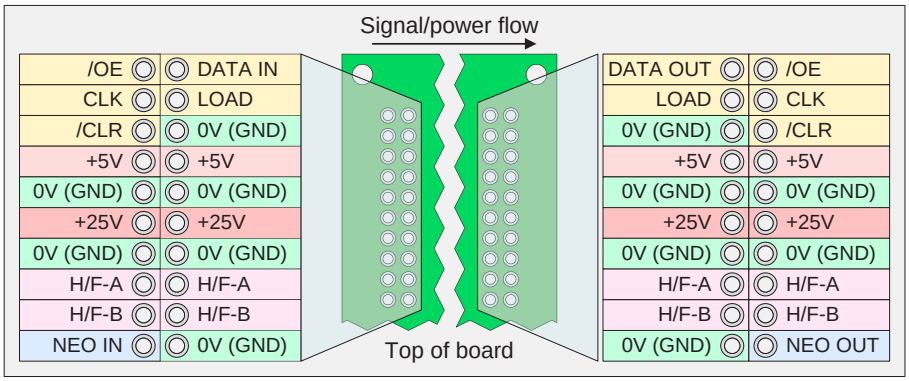


Fig.5: pin connections on the connectors.

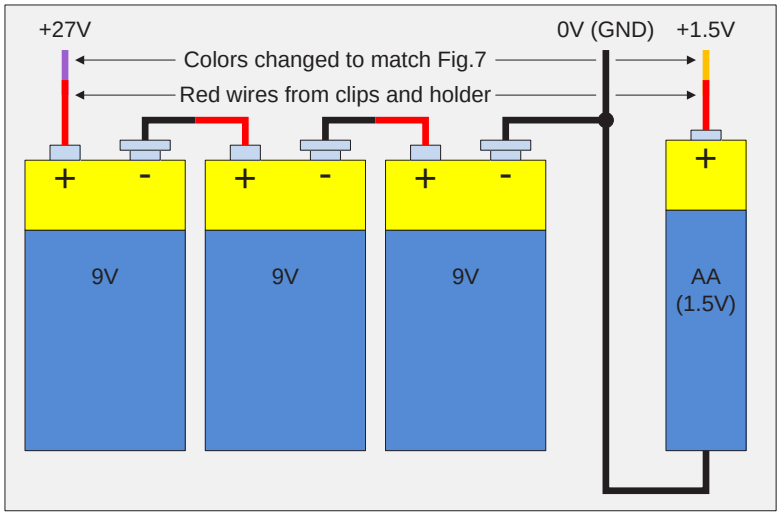


Fig.6: battery connections (holders and clips not shown).

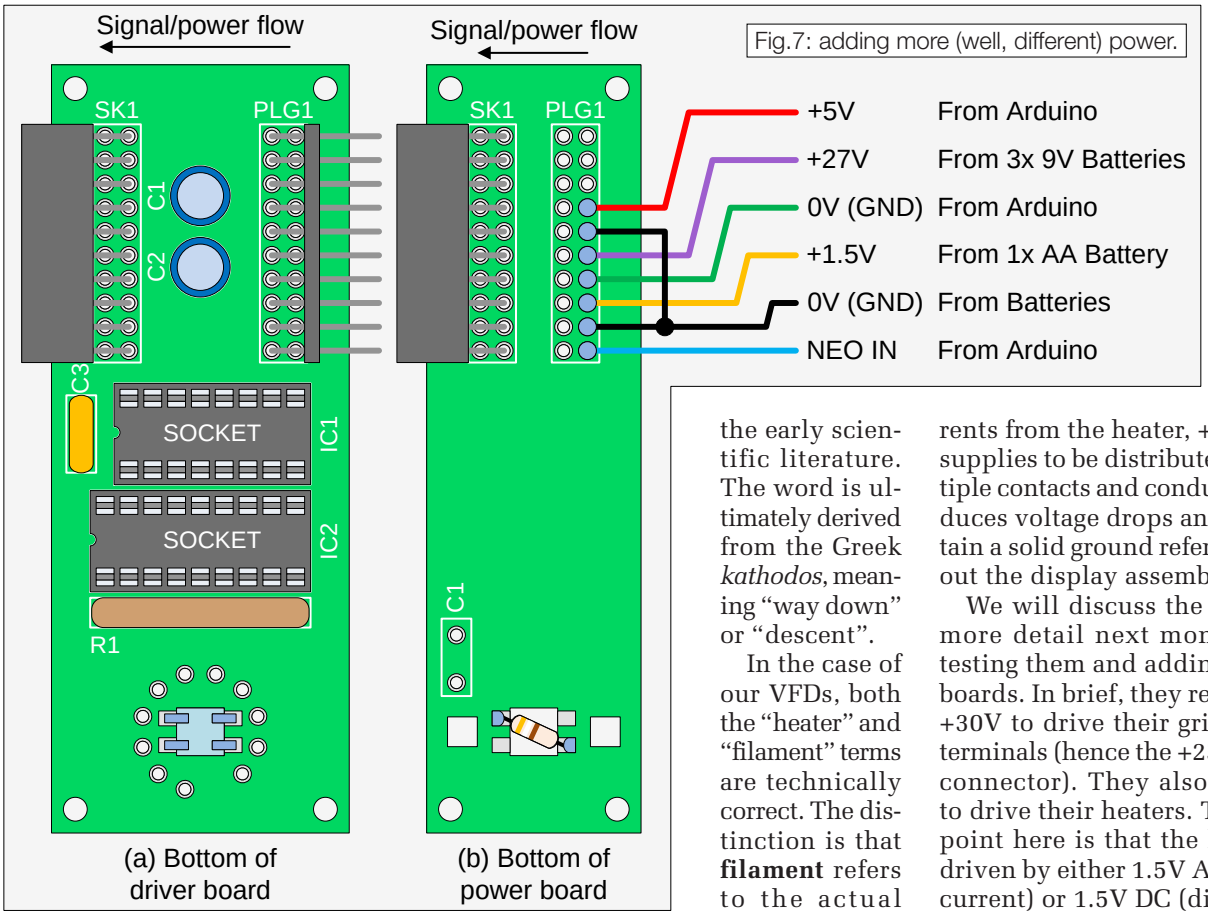


Fig.7: adding more (well, different) power.

wouldn't hurt to use a stiff toothbrush to remove any stray flecks of solder, thereby preventing them from making unwanted connections.

Historical quirks

One question that keeps beginners up at night (well, some beginners… well, me) is why we use "K" (or "k") to indicate the cathode terminal on capacitors, diodes, and LEDs rather than "C" (or "c").

This is one of those delightful historical quirks of electronics. Being English, my temptation is to blame the Germans, but perhaps that would be uncharitable, even though it's partly their fault.

You see, "C" was already widely used for other things, especially "collector" in bipolar junction transistor (BJT) circuits and "capacitor" in schematics and parts lists. More importantly (and of interest to us here because we will be using VFDs), the convention dates back to the days of vacuum tubes (thermionic valves). In tube documentation and circuit diagrams, the electrodes were commonly abbreviated as:

- A = Anode
- K = Cathode
- G = Grid
- H or F = Heater/Filament

"K" was chosen because it starts the word "Kathode", the German spelling of "cathode" that appeared in much of the early scientific literature. The word is ultimately derived from the Greek *kathodos*, meaning "way down" or "descent".

In the case of our VFDs, both the "heater" and "filament" terms are technically correct. The distinction is that **filament** refers to the actual wire inside the tube, while **heater** refers to its function, which is heating the cathode so it emits electrons. Just to keep you on your toes, in the discussions that follow, we will use whichever term seems most appropriate at the time.

Feel the power

Let's start by reminding ourselves of the pin connections on the connectors, as shown in Fig.5. Observe that some pins on each connector are duplicated; that isn't an accident. In some cases, these duplicated pins are primarily for connector symmetry and convenient routing. In other cases, they are intended to be used in parallel to carry higher currents.

For example, according to the datasheet, each IV-11 heater draws around 100mA, so 14 such tubes will consume 1.4A in total. Each connector pin and socket contact has a small but finite resistance. Using two pins in parallel allows the current to be shared between them, reducing the voltage drop and contact heating while providing a little extra margin for reliability.

If you've seen those photos of people with melted connectors on their multi-thousand-dollar graphics processor units (GPUs), you'll understand why this is important!

The same principle applies to the multiple 0V (GND) pins. All of these pins are connected to the board's ground plane, allowing the return currents from the heater, +5V, and +25V supplies to be distributed across multiple contacts and conductors. This reduces voltage drops and helps maintain a solid ground reference throughout the display assembly.

We will discuss the VFD tubes in more detail next month as part of testing them and adding them to our boards. In brief, they require +25V to +30V to drive their grids and anode terminals (hence the +25V pins on the connector). They also require 1.5V to drive their heaters. The important point here is that the heater can be driven by either 1.5V AC (alternating current) or 1.5V DC (direct current).

In our implementation, we will drive the heater using a 1.5V DC supply, with H/F-A connected to +1.5V and H/F-B connected to 0V (GND). However, the board itself does not make this assumption, allowing the heater to be driven from either a DC or AC source.

In the fullness of time, we will create a power supply board that provides +1.5V and +25V to power the VFDs and +5V for the digital logic. For now, we can use our Arduino to provide 5V, a single AA cell to provide 1.5V, and three 9V batteries connected in series to provide 27V, which is smack in the middle of the 25-30V required to drive the VFD's anode terminals.

You can buy a battery case that holds a single 1.5V AA battery on Amazon. You can pick up some 9V battery clips from the same source. The main thing is that everything must have a common 0V (GND) connection, which we can achieve by wiring things as illustrated in Fig.6.

Physical battery clips and holders typically use red and black wires to indicate the positive and negative terminals, respectively. In Fig.6, we've transitioned the colour of the red wires into purple and orange to indicate +27V and +1.5V, respectively. This makes it much easier to understand what's going on when we connect the batteries to our existing power board, as seen in Fig.7.

Note that the black 0V (GND) and

green 0V (GND) signals are connected by the board's ground plane.

Trust but verify

I'm 100% confident that you've attached the components and soldered them to your driver boards perfectly. On the other hand... this is the perfect time to perform a few basic tests on these little scamps before we add the ICs and VFD tubes, especially if one's soldering skills sometimes leave a little something to be desired.

How many (or how few) of these tests you perform is entirely up to you. The more testing you do, the greater your confidence that everything will work as planned, and the less likely it is that your life will become briefly and unexpectedly exciting in the not-too-distant future.

Let's start with the driver board on its own, that is, not plugged into the power board. We'll commence with the top of the board as shown in Fig.8(a). Before reaching for your multimeter, it's worth taking a few moments to clean away any excess flux residue from your recently soldered board.

Doing so offers several advantages, including making it easier to inspect the solder joints, allowing the multimeter's probe tips to make better contact. It also makes the board look nicer, which is a bonus when you show it off to your friends (or your cat).

A small stiff-bristled brush (like an old toothbrush), together with 99% isopropyl alcohol (IPA), will remove most rosin-based flux residues. Wipe the board with a lint-free cloth or paper towel, then allow any remaining IPA to evaporate completely before performing the tests.

Continuity mode is one of the handiest functions on a multimeter, especially when you're assembling PCBs. It's designed to answer one simple question: "Are these two points electrically connected?" In this mode, the meter checks for a very low-resistance electrical path between the two probes. When it detects such a path, it emits a beep.

Try this now. Set your multimeter to continuity mode, then touch the probe tips together and make sure you hear a beep. Is this step really necessary? It is, trust me! You'll feel really foolish if you spend an inordinate amount of time diagnosing a non-existent problem with your board, only to discover that the real problem is your multimeter has a flat battery (don't ask me how I know this).

Now touch the probes to the two GND pins on the male connector. Again, you should hear a beep. If not, check the solder joints on the connector.

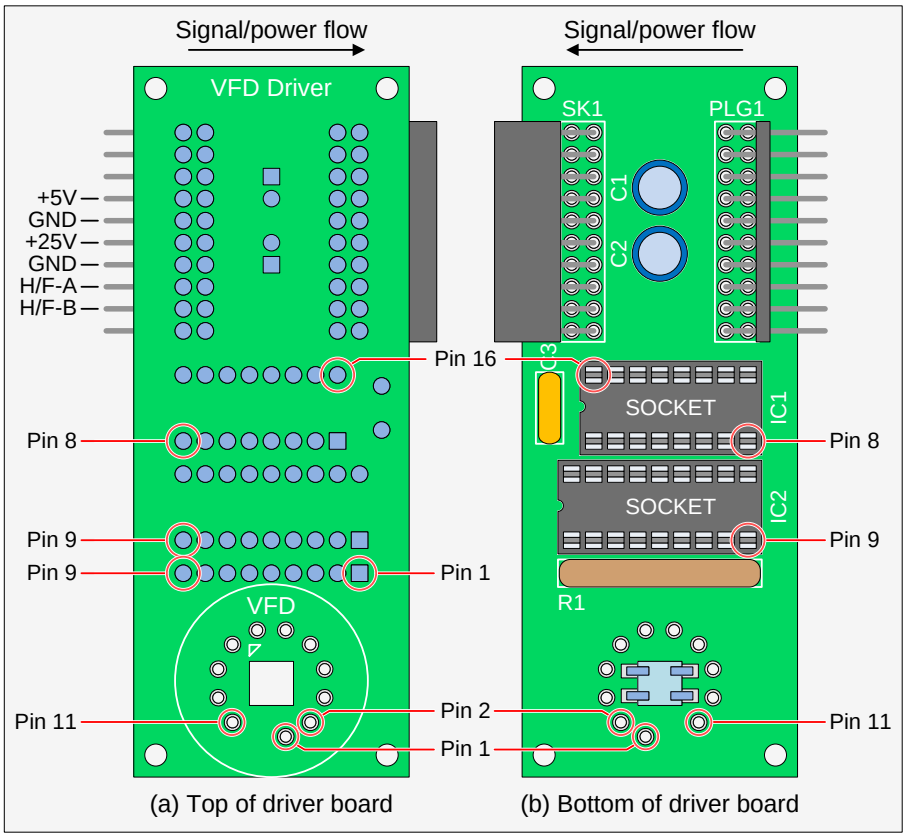


Fig.8: test points for initial verification.

If your meter doesn't support a continuity mode, set it to measure resistance. The resistance between any two connected points on this board should be very low, such as 0.0Ω or 0.1Ω.

If you see a value of "OL" (which stands for "Over Limit", "Open Loop", or "Open Line", depending on what the manufacturer has had for breakfast that morning), this means that the resistance is higher than the meter can measure on the selected range—or, more commonly, that there is effectively an open circuit (infinite resistance).

The following lists some more point-pairs to check (entries like "IC1.8" mean pin 8 on IC1):

- GND to IC1.8
- GND to IC2.9
- +5V to IC1.16
- +25V to R1.1
- +25V to VFD.2
- H/F-A to VFD.1
- H/F-B to VFD.11

If any of these tests fail, it's likely that one or more solder joints need inspection and attention. However, if you've heard nothing but beeps (or seen nothing but extremely low resistance values), it's time to up our game.

Still working on the topside of the board, set your multimeter to resistance mode (select the 20kΩ range if you don't have an auto-ranging device), and measure the resistance between R1.1 and R1.2. If you see an OL value, you either have a bad solder joint, or you've forgotten to solder the resistor pack (hey, it happens).

Assuming you're using a ±2% tolerance resistor pack (like the one I've specified in the parts list), a nominal 10kΩ resistor may measure anywhere between 9.8kΩ and 10.2kΩ, so don't be alarmed if your meter doesn't display exactly 10.00kΩ, because that's perfectly normal.

On the other hand, if you see a value between 19.6kΩ and 20.4kΩ, you have a problem because this indicates that you're measuring two of

Devices and components used in this issue

Qty	Part	Source
15	16-pin IC sockets	www.amazon.co.uk/dp/B0BZYK4DV6
15	18-pin IC sockets	www.amazon.co.uk/dp/B07H3SMWXZ
15	9-pin SIP resistor network	Mouser Part# 652-4609M-1LF-10K
30	63V 1µF radial electrolytic capacitors	Mouser Part# 710-860020772005
1	AA battery holder	www.amazon.co.uk/dp/B00H8U9YJI
3	9V battery connector clips with wires	www.amazon.co.uk/dp/B083QFFH66

your 10kΩ resistors in series. There are two possible explanations. The first, which really isn't a problem at all, is that you have inadvertently placed your probes on R1.9 and R1.8 instead of R1.1 and R1.2, in which case you're a silly sausage.

The second, which **is** a problem, is that you've inserted the resistor pack the wrong way round. Fortunately, you've discovered this at the ideal time to practice your desoldering skills (YouTube is your friend; there are plenty of soldering and desoldering tutorials hosted there).

Now it's time to connect the driver board to the power board, connect the batteries and Arduino to the power board, as in Fig.7(b), and perform some simple tests on the bottom of the driver board, as illustrated in Fig.8(b).

In this case, we're using the Arduino only to provide +5V power. However, if our earlier NeoPixel test is still running, don't be alarmed if you see the NeoPixel flash occasionally.

Set your multimeter to its DC voltage mode (select the 20V range if you don't have an auto-ranging multimeter).

Place the black probe on IC1.8 or IC2.9; either provides a convenient 0V (GND) connection. Leave this probe in place while touching the red probe to IC1.16. You should see approximately +5V.

Leave the black probe where it is and move the red probe to VFD.2. You should now see a reading of approximately +27V (remember, we're currently using 3 × 9V batteries in lieu of the +25V supply we will eventually employ).

Finally, move the black probe to VFD.11 and the red probe to VFD.1. In this case, you should see our heater voltage of 1.5V.

Now we're really cooking on a hot stove. All you have to do is verify that your remaining driver boards are ready for action in preparation for what is to come.

I still have no idea why they refer to me as "Mad Max"!

Why all the fuss?

You may be wondering if all this testing is necessary for such a simple board. Like all of these columns, the real goal isn't simply to build projects like our Chronological Contraption. It's also to learn tips, tricks and techniques, acquire good habits, and develop the sort of troubleshooting skills that will stand you in good stead in the future.

As projects grow larger and more complex, finding a fault after everything has been assembled can become difficult. Spending a few minutes verifying each stage as you go almost always pays handsome dividends later.

You won't truly appreciate the wisdom of my words until you skip these checks and then spend several hours wondering why nothing works! You may subsequently discover something like a tiny solder bridge that would have taken just a few seconds to find and fix before any of the more complex, delicate, or expensive components, such as ICs and tubes, were installed and possibly damaged by an over-voltage condition.

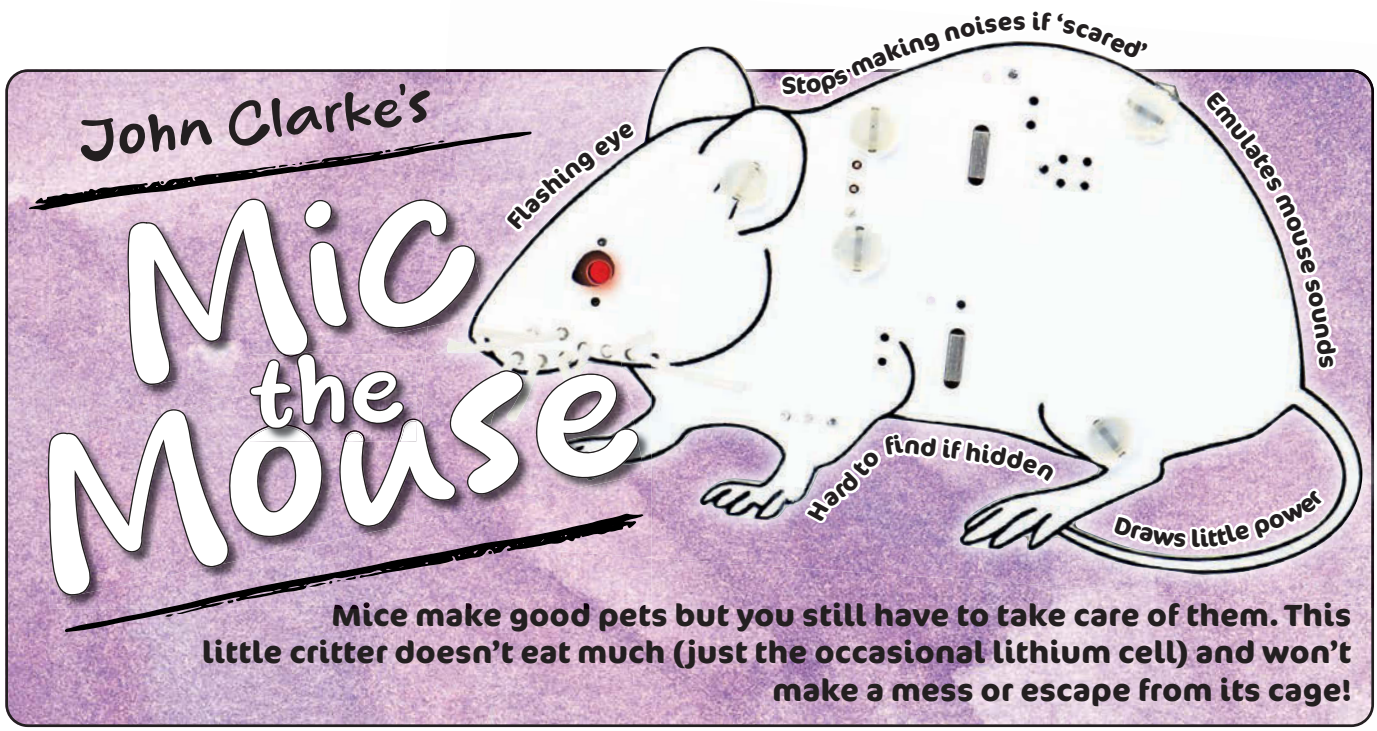
Next time

In our next column, we'll learn a lot about VFD tubes, including how to test them **before** we solder them into our driver boards (this is definitely the preferred sequence of operations).

After that, we'll add the tubes to the boards, along with our two ICs. We'll practice displaying values on the tubes while feasting our orbs on their glorious blue-green, retro-futuristic fluorescent glow.

As always, If you have any thoughts you'd care to share on anything you've read here, or questions for me, please feel free to drop me an email at max@clivemaxfield.com PE

HAMMOND®
1551W IP68 miniature enclosures
Learn more: hammondmfg.com/1551w
uksales@hammondmfg.com
01256 812812


John Clarke's
Mic the Mouse
Flashing eye
Stops making noises if 'scared'
Emulates mouse sounds
Hard to find if hidden
Draws little power
Mice make good pets but you still have to take care of them. This little critter doesn't eat much (just the occasional lithium cell) and won't make a mess or escape from its cage!

Built on a mouse-shaped printed circuit board (PCB) and using relatively few parts, Mic the Mouse is ideal for fun and can be used to play pranks on family and friends. Mic the Mouse only produces squeaking sounds when everything is quiet. Make a noise, and he goes quiet. This makes him difficult to locate. He will start squeaking again, but only after a period of silence.

Mic the Mouse (Mic for short) is best described as mousy coloured and mousy shaped. To further add to the realism, there is provision for whiskers. He sits vertically, with a slight lean backwards, and is supported at the rear using a stand that attaches to the back.

Elephants and mice have been known in folklore and animations to have a unique relationship (see https://pemag.au/link/ac63), which is why the rear stand is reminiscent of an elephant's rear end. Or perhaps it's just because that's a suitable shape to hold the mouse up. Nobody knows what happened to the front half of the elephant – except maybe Mic!

It is also well-known that mice have a unique relationship with humans. The poem by Robert Burns, entitled "To a mouse", begins: "The best laid plans of mice and men, often go awry and leave us nought but grief and pain for promised joy". This couldn't be more true with Mic the Mouse.

Hunt him down you try, but Mic is elusive. He won't reveal his whereabouts if you make any noise. But wait quietly and he will begin his merry squeaks again. Hide Mic in a cupboard, on a shelf, or simply in plain sight, and have others become horrifically aware that there is a mouse in the house. But where? You may be confronted by Mic's eye flashing in a terrifying manner.

Previous designs

Way back in August 1990, Silicon Chip magazine published an electronic cricket called Horace (siliconchip.au/Article/6925). Horace was similar to Mic the Mouse, except it chirped cricket sounds rather than producing mouse sounds. It only chirped when there was quiet, and was quiet himself when there was ambient noise.

It utilised an electret microphone, quad op amp IC and a piezo transducer. This was powered by a 9V battery and its current draw was 3mA. With a 600mAh capacity, Horace could run for about 200 hours or about 8 days continuously. The reason for the "Horace" name has been lost in time.

We published the Silicon Chirp cricket design in April 2024. This design used a low-power microcontroller to offer much longer running times.

The reason for Mic's name is a bit more straightforward. Firstly, it keeps up the tradition of alliterative names for animal characters like Dorothy the Dinosaur, Peppa Pig and so on. The other reason is that Mic uses a microphone to listen for sounds.

To make Mic the Mouse a similar size to a real mouse, we need to use a few tricks. A 3V lithium cell is much smaller than a massive (by comparison) 9V battery. However, a 3V cell does not have as much capacity (200mAh, 600mWh) compared to a 9V battery (600mAh, 5.4Wh), so the current consumption needs to be kept as low as possible.

To get a reasonable cell life, we need to reduce the current consumption from the 3mA of Horace the cricket down to at least 1mA to get the same cell life. However, we can do a lot better than that; we reduced the current draw to an overall average of 105µA. That's a reduction of around 28 times compared to Horace!

This was achieved by using a low-power microphone and a microcontroller to mini mice (sorry, minimise) the current draw by only powering the microphone when required. Also, we only flash the LED eye momentarily to mini mice the power used.

Like a real mouse, it sleeps to reduce its current consumption to an absolute minimum; it is woken up periodically by a timer. It's a bit like hibernation, a trick Mic has stolen from bears.

The low current microphone we use is a MEMS (micro-electromechanical systems) type, as used in phones. It is supplied in a tiny package that measures just 2.75 × 1.85 × 0.95mm and requires reflow soldering as the contacts are underneath the package. That makes it difficult to hand-solder.

Coin cell warning!

This project contains a small lithium 'coin' cell that represents a serious health risk should the cell be swallowed by a child. Young children are most at risk. Ensure that the cell is kept secured using the cell capture screw and nylon spacer and that it is tightened fully to prevent undoing by hand. Keep this project away from small children. Also, keep unused cells in a secure place away from children, such as in a locked medicine cupboard.

New cells should be kept within the original secure packaging that requires scissors to open until required for use.

If you have any older button/coin-cell powered devices that provide easy access to the cells, store them in a safe place when not in use. Alternatively, devise a method to make the cell access more difficult, such as gluing the cell compartment shut so that a child can't open it.

Fortunately, the MEMS microphone is available pre-soldered on an inexpensive module, which also includes an amplifier and 3V regulator.

Circuit details

Mic's complete circuit is shown in Fig.1. It's based around microcontroller IC1, a PIC16F15214-I/SN, powered by a 3V lithium cell (BAT1). Power is applied via a slider switch. Mic does not draw much current, typically only about 0.36µA while asleep. This rises to around 1mA when monitoring for ambient sounds and 1.6mA while making noise.

Diode D1 is included as a safety measure to prevent damage to IC1 should the cell be inserted incorrectly. The cell holder doesn't stop you from inserting the cell with the incorrect orientation (it should be positive side up).

With the positive side down, the cell will be shorted out by contact with the sides and top spring contacts. However, during insertion, there could be a brief period when there is no contact with the cell holder sides, so the circuit could be supplied with a reversed polarity voltage that could damage IC1. In that case, D1 clamps this voltage to a low level.

The cell will lose some of its capacity if left connected in reverse for more than a few seconds, but that's better than damaging the chip.

IC1 is clocked by an internal 4MHz oscillator. Its power supply pin is bypassed with a 1µF capacitor. IC1's job is to supply power to and monitor the MEMS microphone module (MIC1) output, drive the piezo transducer to make mouse sounds, flash the LED used for Mic's eye and also check if S2 is pressed.

The MEMS microphone module (MIC1) is powered via IC1's RA4 digital output, which goes high (near to the cell voltage) when required. When powered, the A output on the module (pin 3) provides an amplified signal from the MEMS microphone.

The circuit of the microphone module that includes the MEMS microphone is shown within the dashed box in Fig.1. Its onboard regulator, U1, supplies 3V to the MEMS microphone itself (U3) and provides a bias voltage to pin 1 of the op amp, U2. U1 is a low-dropout regulator, so with a 3V input, its output won't be much below that.

The output from the MEMS microphone (U3) is amplified by op amp U2, which is configured with a gain of 50. The non-inverting input is held at half supply using the two 10kΩ divider resistors across the 3V supply. So, with a 3V supply to the MEMS module, the DC output from U2 is typically at 1.5V. A MEMS module output signal when subject to noise is shown in Scope 1.

For data on the MEMS microphone and module, see https://pemag.au/link/ac64 and https://pemag.au/link/ac65

The current consumption of the MEMS module is typically 287µA at 3V. That's the total of U1 (7µA typical, 15µA maximum), U2 (80µA typical, 185µA maximum) and U3 (50µA typical, 150µA maximum). The voltage divider comprising the two 10kΩ resistors in series across the 3V supply also contributes 150µA. We measured our module's draw as 330µA.

The PIC16F15214 (IC1) monitors the microphone signal using its AN5 analog input. We have AC-coupled MIC1's output to that pin using a 100nF capacitor and biased the voltage to 0V via a 10kΩ resistor. This has the signal at the AN5 input (pin 2) normally

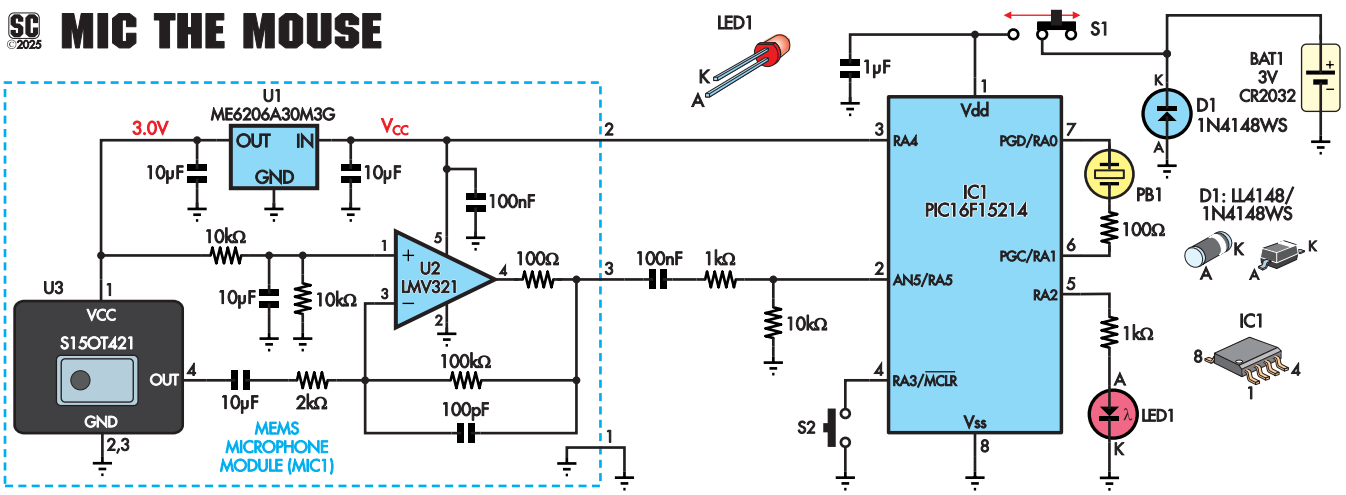
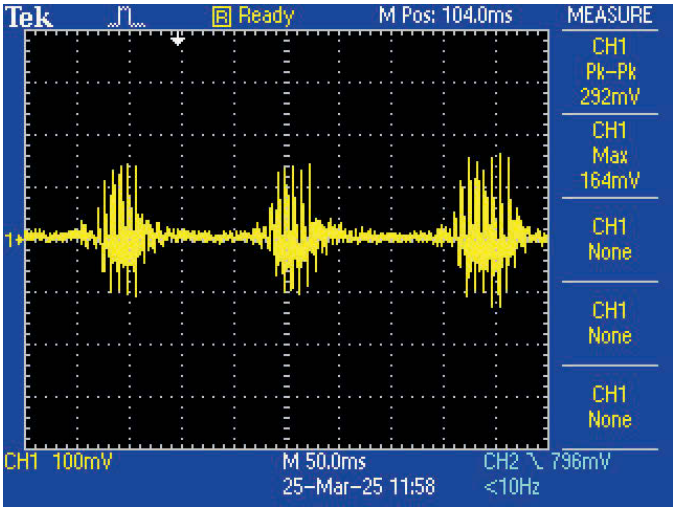
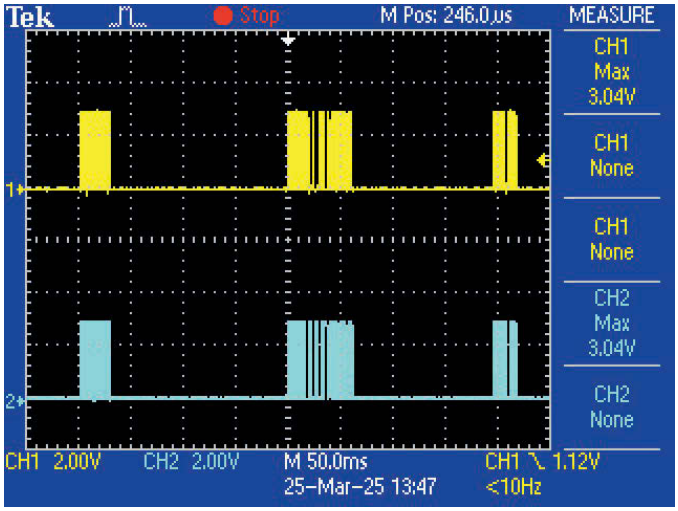


Fig.1: the components inside the dashed cyan box are on the MEMS microphone module. IC1 monitors its output and determines when to flash the eye LED and create squeaking noises using the piezo sounder.



Scope 1: this oscilloscope trace shows the output from the MEMS microphone module after AC coupling to the AN5 input (pin 2) of IC1. The signal level goes above 100mV peak.



Scope 2: the RA0 (yellow) and RA1 (cyan) output waveforms when producing mouse sounds. The signal bursts vary randomly in length, with variable periods of silence in between. It isn't obvious from this that the signal frequencies and duty cycles also vary.

sitting close to 0V and swinging above and below that.

A diode clamp internal to pin 2 of IC1 will limit negative excursion to -0.3V, while the 1kΩ series resistor limits the current in the clamping diode.

We do this so that pin 2 sits near 0V with no signal (ie, silence). Also, while the 10kΩ/10kΩ resistive divider in the module theoretically causes the signal to sit at exactly half the supply voltage, the supply voltage can vary because it's coming from a microcontroller output which has a fairly significant output impedance of around 116Ω.

That means that supply to the MEMS module can be anywhere between about 33-58mV less than the IC1 supply due to the voltage drop at the RA4 output. The MEMS module current draw also varies, so it is difficult to predict the MEMS module's idle output voltage with sufficient accuracy to allow for threshold detection of any small signal that is superimposed on it. Re-biasing the signal to 0V solves this difficulty.

Noise detection

To detect ambient noise, we convert the voltage at the AN5 input into a 9-bit digital value every 1ms (1000 times per second). The digital value ranges from 0 to 511. If this exceeds a specific threshold value, it is detected as noise. This threshold can be adjusted between 1 and 10 in ten steps, corresponding to an analog range of about 12-65mV (assuming a 3V supply).

Lower threshold settings give Mic a greater sensitivity to noise. More on this later.

Mouse sounds

IC1's RA0 and RA1 output pins drive the piezo transducer to produce the mouse noises. The piezo is driven in bridge mode by the two outputs, increasing the AC voltage across it to produce a louder sound.

When the RA0 output is high, RA1 is low and vice versa. In one condition, there is +3V across the piezo transducer, and in the other, -3V. This results in a 6V peak-to-peak square wave. A 100Ω resistor limits the peak current into the transducer's capacitive load as the outputs switch.

The mouse sounds comprise various frequency bursts with variable length gaps in-between. The signal frequency varies between bursts and also within each burst. Scope 2 shows three such bursts.

While not visible in Scope 2, there is considerable detail within each signal burst. At the beginning of each burst, the duty cycle starts off quite low. This means the piezo transducer is driven only with brief pulses, resulting in a low volume. As the duty cycle increases, the output from the piezo transducer also increases. The duty cycle is increased a little each cycle until it reaches a 50%.

A similar change in the duty cycle occurs at the end of each burst. The duty cycle is reduced on each cycle until it reaches zero, so that the volume falls back to zero. This gives the signal bursts soft starts and soft finishes, preventing loud clicking sounds from being produced by the piezo transducer at the beginning and end of each burst.

We also use lower duty cycles to reduce the volume level within each burst instead of having a constant level. A varying volume level sounds more natural.

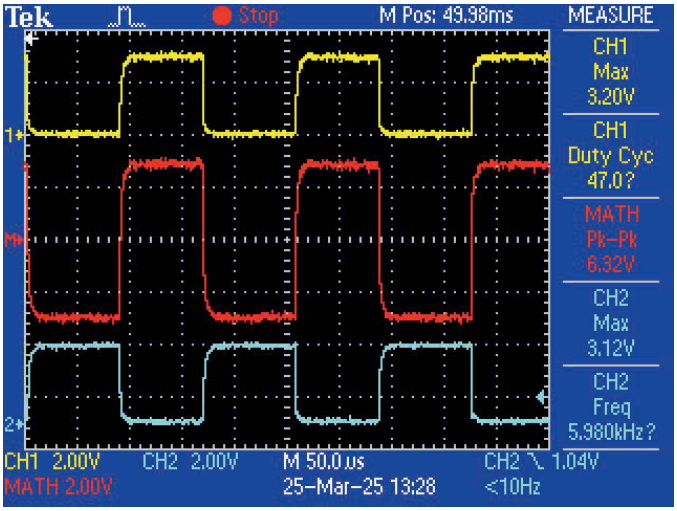
The greatest volume available from the piezo transducer is when it is driven at 50% duty, as shown in Scope 3. The push-pull drive from the RA0 and RA1 outputs is visible too. This is necessary to provide a sufficient sound level from a supply voltage of just 3V or less.

The drive frequency also varies over the burst period. If we were to just have the same frequency throughout, it would sound just like bursts of a single tone, like Morse Code. By having a frequency mix, the bursts sound considerably less electronic in nature.

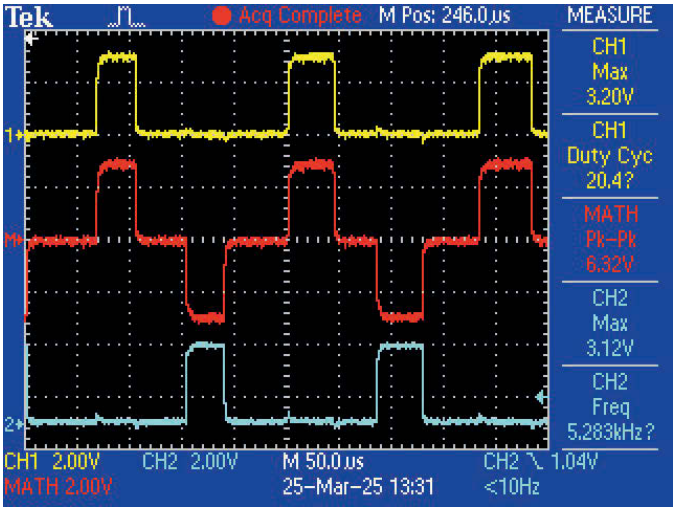
Mic's eye

LED1 is driven via the RA2 output via a 1kΩ current-limiting resistor. This LED is made to flicker when Mic is producing sound. The LED is also used to indicate the threshold level used to detect ambient noise during the setup process. It flashes between one and ten times to indicate the chosen threshold value. The LED also flashes briefly when Mic is powered up to acknowledge this.

Pushbutton S2 is used to set this threshold. IC1 detects that S2 is closed



Scope 3: an expanded view of the drive to the piezo transducer, showing how the ~3V peak square wave signals from RA0 & RA1 (yellow and cyan) combine to form a 6V peak-to-peak square wave across the transducer (red trace). The duty cycle here is near 50%.



Scope 4: this is similar to what's shown in Scope 3, except the duty cycle is lower, at around 20%. This reduces the output sound level.

by monitoring its pin 4 digital input (RA3). When S2 is pressed (closed), the voltage is close to 0V. When the switch is open, an internal pullup current in IC1 keeps the RA3 input high. The S2 switch closure is only checked during power-up; if it's low (closed) then, the threshold setup process starts.

Power control

Much of the design work went into minimising the current draw from the small 3V cell. Shutting down the circuit is the major way to do this. When IC1 is in sleep mode, its oscillator is off and the power supplies to the MEMS module and LED are also switched off.

A separate 'watchdog' timer starts running in sleep mode, to wake IC1 periodically. This varies between 4, 8, 16, 32, 64, 128 and 256 seconds in a randomised order.

To extend the sleep periods and save more power, IC1 is sent back to sleep immediately upon waking 30 times. This provides an off-time between two minutes (when there is a 4s watchdog timeout) and about two hours for the 256s timeout. During this period, the current consumption is very low; we measured this at 0.36µA with a 3V supply.

IC1 itself draws just 0.9µA in sleep mode, including the watchdog timer and oscillator current draws.

After these 30 sleep periods, IC1 powers up the MEMS microphone module and checks for ambient sound. During this period, its current consumption is about 1mA. This is mainly due to the MEMS module consumption at about 330µA, and IC1 drawing around 660µA while running.

There needs to be a 2-16 minute quiet period (again a randomised value; it's either 2m, 4m, 8m or 16m) before it is deemed to be quiet enough for the mouse to make noises. Should noises be detected during the listening period, IC1 will go back to sleep for another randomly chosen sleep period.

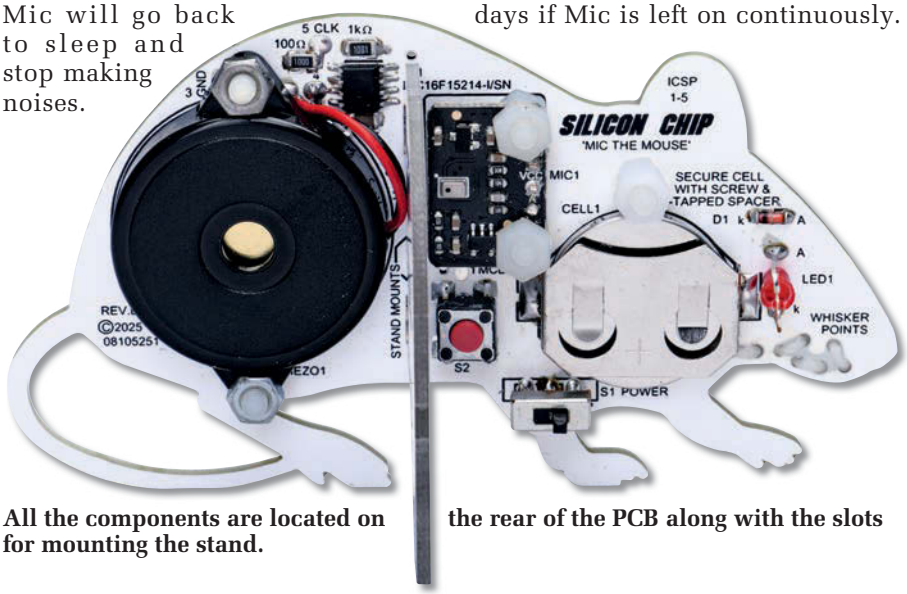
If no sound was detected, Mic the Mouse will begin to make mouse sounds. During this time, his current consumption is around 1.6mA. These sounds run over a variable-length period between 100ms and two minutes; a typical duration is around 30 seconds. If noise is detected in between making the mouse noises, Mic will go back to sleep and stop making noises.

There is a brief 5ms delay between each mouse sound ceasing and the beginning of monitoring ambient noises at the AN5 input. This wait is to prevent the MEMS microphone from picking up sounds from the piezo transducer.

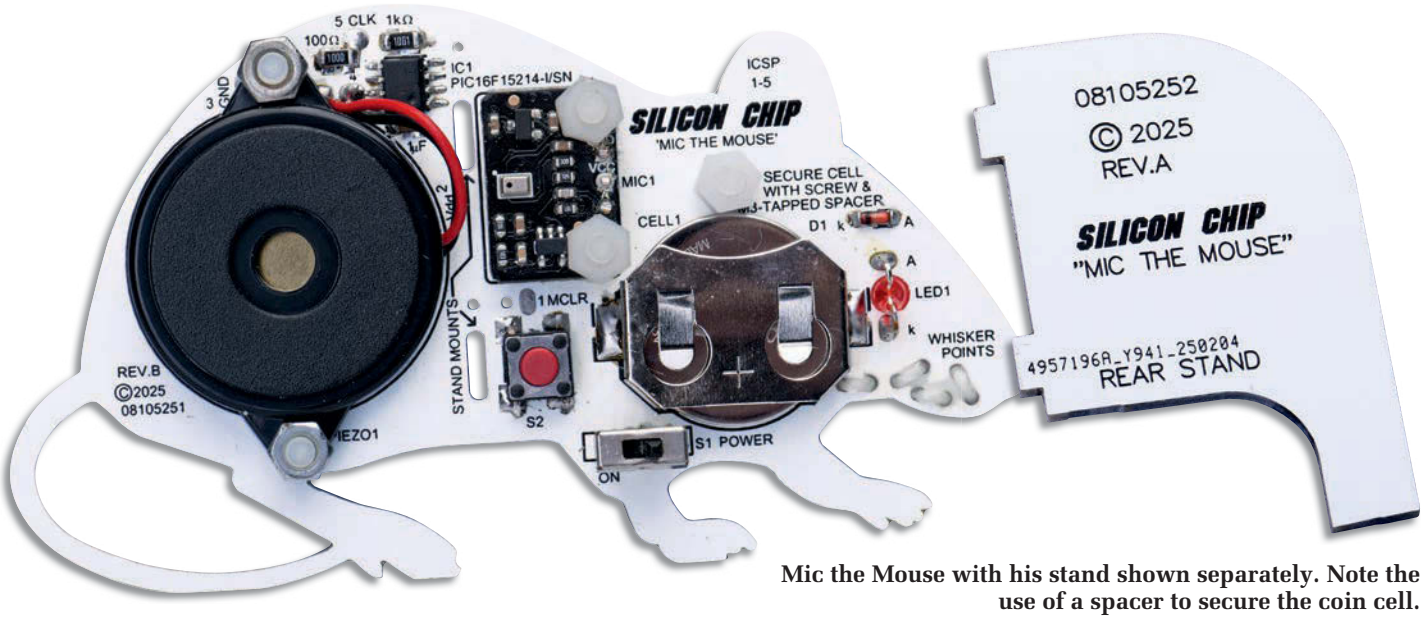
Adding up the total current draw taking into account the typical sleep, checking for ambient sound and the mouse sounds operation periods, we estimate the overall current draw is an average of 105µA.

It is checking for ambient noise (drawing 1mA) around 9% of the time, making mouse sounds (1.6mA) around 1% of the time and in sleep mode (0.36µA) 90% of the time.

Considering a typical 3V cell has a capacity of 200mAh, we expect Mic the Mouse to operate on the one cell for around 1905 hours. That's 79 days if Mic is left on continuously.



All the components are located on the rear of the PCB along with the slots for mounting the stand.



Mic the Mouse with his stand shown separately. Note the use of a spacer to secure the coin cell.

If power is switched off, the current draw from the cell becomes close to zero, with the only draw being cell leakage and diode D1's reverse leakage. These are very low and in the nanoamps (nA) region.

If you handle the cell with your fingers across the insulating ring between the positive and negative contact areas, the leakage current can be higher due to skin oils bridging the terminals. Cleaning the cell with methylated spirits or similar will prevent this extra leakage from occurring.

Construction

The parts for Mic the Mouse fit on a double-sided plated-through PCB coded 08105251, measuring 96 × 53mm, with a white solder mask and black labelling.

The rear stand plugs into the component side of the Mouse PCB to support it; it is also a PCB, coded 08105252, that measures 48 × 31mm. The main component overlay diagram is shown in Fig.2.

Check that the tabs on the stand fit into the Mouse slots before assembly. If it is difficult to fit the two together, a small amount of filing may be necessary. The stand should be removed while installing parts on the Mouse PCB.

If you are going to use countersunk screws, the front of the PCB will need its holes countersunk so that the screw heads fit neatly, almost flush with the PCB face.

Begin by installing the microcontroller (IC1), which comes in an 8-pin SOIC SMD package. You will need a soldering iron with a fine tip, a magnifier and good lighting. First, place the chip with its pin 1 locating dot to the lower right and with the IC leads aligned with the pads. Then solder a corner lead and check that it is still aligned correctly.

If it needs to be realigned, remelt the soldered connection and move the IC to align it again. When correct, solder all the remaining pins. Any solder that runs between the IC pins can be removed with solder-wicking braid (ideally with the aid of a little flux paste).

Continue by installing the resistors. These will have value codes printed on them, with the last number indicating how many zeroes follow. For the resistors used, the codes will be 101, 100R or 1000 for 100Ω, 102 or 1001 for 1kΩ and 103 or 1002 for 10kΩ.

Two resistors and one capacitor are located beneath the MEMS microphone module, so these need to be

Parts List – Mic the Mouse

1 double-sided plated-through white PCB coded 08105251, 96 × 53mm
1 double-sided-plated through white stand PCB coded 08105252, 48 × 31mm
1 Fermion MEMS microphone module (MIC1)
1 30 × 5.5mm passive piezo transducer (PB1) **[HYR-3006/AT3040]**
1 SIL SPDT mini vertical slider switch (S1) **[SS12D00G3]**
1 4-pin 6.2×6.5mm tactile switch (S2) **[SKHMQME010 or similar]**
1 CR2032 surface-mount folded phosphor bronze PCB mount cell holder (BAT1) **[BAT-HLD-001 or similar]**
1 CR2032 3V lithium cell
1 3-pin header, 2.54mm pitch (usually supplied with the MEMS microphone)
1 260mm length of white 0.8mm diameter bamboo cord
5 M3 × 10mm nylon or polycarbonate screws (cheese or countersunk head)
4 M3 nylon or polycarbonate hex nuts
2 M3 nylon, polycarbonate or metal hex nuts
1 M3 × 6.3mm tapped nylon standoff/spacer

Semiconductors
1 PIC16F15214-I/SN 8-bit micro programmed with 1810525A.HEX (IC1)
1 SMD 75V 500mA fast signal diode, such as 1N4148WS or LL4148 (D1)
1 3mm standard brightness red LED (LED1)

Capacitors *(all SMD M2012/0805 or M3216/1206)*
1 1µF 50V X7R
1 100nF 50V X7R

Resistors *(all SMD M2012/0805 ⅛W or M3216/1206 ¼W)*
1 10kΩ 1%
2 1kΩ 1%
1 100Ω 1%

SC Kit (SC7508, ~£24 + P&P): *has everything but the CR2032 cell (see p78)*

installed before the MEMS module is in place. The 100nF and 1µF capacitors can be soldered in next; their orientations do not matter. These will not be marked with values, but the 1µF capacitor is likely to be thicker than the other.

Diode D1 can now be installed, taking care to orientate correctly. There is sufficient tinned copper area to allow MiniMELF/SOD-80 or SOD-323 package devices to be soldered in.

S1 is a through-hole slide switch but you should insert its pins into the allocated holes high off the PCB so the leads don't protrude through to the other side of the PCB. You can then solder the switch pins to the top side of the pads, not the underside, keeping the visible side of Mic unmarred by solder joints. The on-position for this switch is marked on the silkscreen.

Switch S2 is surface-mounting tactile pushbutton switch, so solder its four corner pins to the pads.

The two mounting holes on the MEMS module need to be drilled out to 3mm to allow the module to be raised off the PCB using nuts as spacers, and secured with M3 machine screws and nuts. The MEMS microphone module is connected electrically using a standard 3-pin 0.1-inch/2.54mm pitch header.

Solder this header initially on the component side of the mouse PCB, with the lead ends flush with the non component side, like with the slide switch. After that, slide the black plastic spacer off the pins.

Before soldering the MEMS microphone module, attach it to the mouse PCB using M3 nylon or polycarbonate screws and nuts, with the screw heads on the non-component side and one nut securing the screw to the PCB on the component side. The MEMS module is then placed on the screws and two more nuts added to hold the module in place.

Do not use metal nuts as they could cause short circuits. With the module attached with the screws, you can then solder the three pins to the pads on the MEMS microphone module.

The cell holder is mounted with the cell entry side towards the mouse's ears. That allows the cell capture screw to keep the cell in place, preventing small children from removing it. This complies with European Standard EN 71-1:2026 on toy safety.

While Mic the Mouse is not really a project for very young children, it could be used in a household where young children live or visit, who could potentially swallow button cells if they find one and manage to remove it.

For our project, the cell is held within a compartment, with the exit blocked by a 10mm M3 screw that is inserted from the non-component side of the PCB and secured on the cell holder side with a 6.3mm-long nylon tapped standoff. When tightened with a screwdriver, the standoff cannot easily be removed by hand. An alternative to the standoff is two M3 nuts, with the top one used as a lock nut.

The cell holder is a half-shell type; its metal contacts the positive side of the cell. A tinned copper area on the PCB completes the cell holder, providing the negative connection to the cell.

LED1 is a leaded component, with its leads bent so that they are U-shaped, returning past the LED body. The LED's lens is positioned over the mouse's eye hole; it does not protrude through the hole fully. Solder the leads from the component side and make sure the (longer) anode lead is soldered to the pad on the PCB marked "A".

The wires for the piezo can be soldered to the PCB (the positions are marked 'piezo'). The wires will need to be cut shorter than supplied. The wires will probably be red and black, but it does not matter which colour wire goes to which PCB pad. Typically, including in this case, the transducer is not used as a polarised component.

You will need to drill the mounting holes on the piezo unit out to 3mm to suit the M3 screws. The piezo transducer is then secured with two 10mm-long M3 screws and two nylon, polycarbonate or metal nuts.

Now insert the CR2032 cell into its holder, secure it with the screw and M3 tapped standoff and switch on the power with switch S1. If all is well, the eye LED will momentarily flash to acknowledge power has been connected. The eye also very briefly flashes at the end of each sleep cycle.

Programming IC1

That test assumes IC1 has already been programmed, which it will be if you order it from the Silicon Chip shop, by itself or as part of a kit. If you need to program the PIC yourself, the firmware (1810525A.HEX) can be downloaded from siliconchip.au/Shop/6/2698

If the chip has already been soldered to the board, but is unprogrammed, you will need to wire up a programming adaptor to the PCB, such as a PICkit. Since there is no in-circuit serial programming (ICSP) header, you will need to make the five

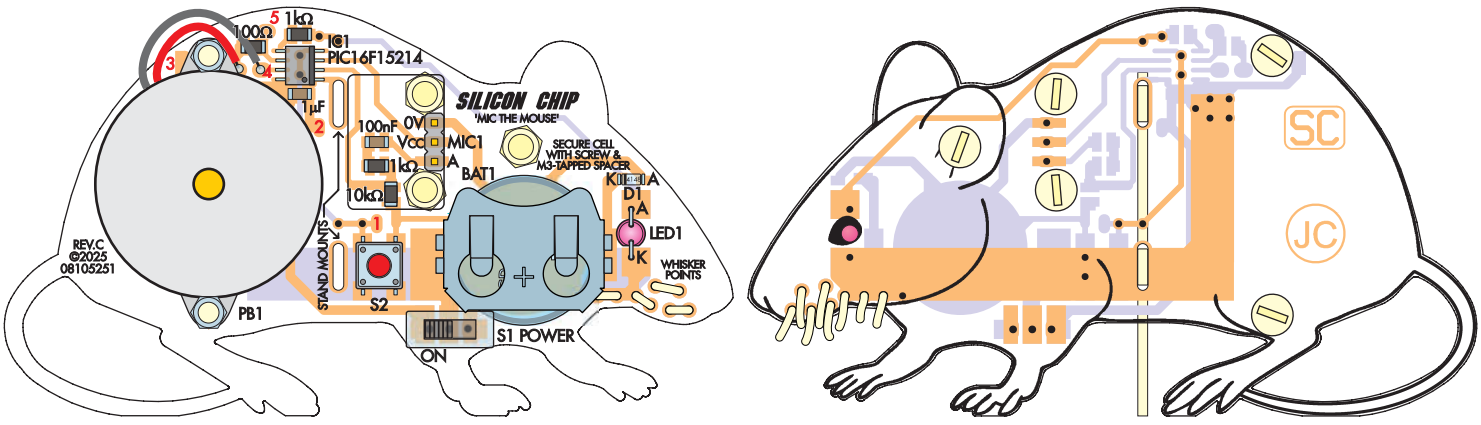


Fig.2: there are about 14 different components mounted on the PCB; don't miss the three that are under the MEMS microphone module. The five pads numbered 1-5 in red are the points you can solder wires to for in-circuit programming of IC1. They correspond to pins 1-5 of a PICkit programmer or similar.

connections separately. They go to pads marked 1-5 on the PCB and in Fig.2; these correspond to the pins on the PICkit programming header (1 = MCLR etc).

Sensitivity to sound

As mentioned previously, sensitivity to ambient sound can be adjusted so that you can select the sound level that Mic reacts to, over a range of 1-10. Lower values provide higher sensitivity to sound, ie, Mic will detect lower noise levels. Higher values mean less sensitivity, so more noise is required to silence Mic.

To adjust sensitivity, switch it on using S1 while holding down S2. This initiates the adjustment mode, where Mic's eye blinks out the sensitivity setting. There is one blink for each sensitivity level. You can test each sensitivity level after the flashes have finished; you have up to 16 seconds to test each level. The eye will flash in response to your making noises.

If the eye continuously flashes due to the detection of background noises, the setting is too sensitive, and a higher value should be selected.

To change to the next sensitivity level, press S2 before the 16 seconds are up. This will cause the eye to flash out the next sensitivity level. You can then test this sensitivity level for up to 16 seconds. Once the sensitivity value has reached 10, the next value will be 1 again.

The selected sensitivity is stored in flash memory, and will be remembered if the power is switched off.

If you wait out the 16 seconds after releasing S2, Mic will begin to make squeaking sounds. This is a quick way to have Mic make some sounds for testing. While making these sounds, Mic also checks whether there is ambient sound. If detected, any mouse sounds will cease, causing him to go to sleep.

On a normal power-up without S2 pressed, mouse sounds will begin after about four minutes from switch-on. This period will also depend on whether there is ambient noise present that would prevent Mic from sounding. Further mouse sounds could occur up to two hours later.

Adding some whiskers

The whiskers are made using white 0.8mm bamboo cord. The whiskers can be up to about 30mm in length, so cut each length of cord to about 65mm, allowing two whiskers to be formed by folding the length in half. Then insert each end into two adjacent holes in the whisker region, from the component side of the mouse.

Coat the cords with a thin smear of PVA glue so that they will become stiff when dry. You will need to orientate the whiskers by having the mouse body supported on a stand so that the PCB sits horizontally, with the whiskers hanging downward until the glue is dry.

Finally, the rear stand can be attached at the component side with its two protrusions placed into the slots on mouse PCB. The piezo wire leads will add some holding force to keep the stand in place.

Modifications

If you want to reduce the volume of the mouse squeaks, increase the resistance of the resistor in series with the piezo transducer. Increasing it from 100Ω to 1kΩ will reduce the apparent volume by about 50%. Higher values will provide an even lower volume level. PE

ETI BUNDLE (2) Teach-In 6, 7 and 8 – all on CD-ROM – only £18.95

ELECTRONICS TEACH-IN 6 – CD-ROM
A COMPREHENSIVE GUIDE TO RASPBERRY Pi
Mike & Richard Tooley

Teach-In 6 contains an exciting series of articles that provides a complete introduction to the Raspberry Pi, the low cost computer that has taken the education and computing world by storm.

This latest book in our *Teach-In* series will appeal to electronics enthusiasts and computer buffs who want to get to grips with the Raspberry Pi.

Teach-In 6 is for anyone searching for ideas to use their Pi, or who has an idea for a project but doesn't know how to turn it into reality. This book will prove invaluable for anyone fascinated by the revolutionary Pi. It covers:

- Pi programming
- Pi hardware
- Pi communications
- Pi Projects
- Pi Class
- Python Quickstart
- Pi World
- ...and much more!

The *Teach-In 6* CD-ROM also contains all the necessary software for the series, so that readers and circuit designers can get started quickly and easily with the projects and ideas covered.

ELECTRONICS TEACH-IN 7 – CD-ROM
DISCRETE LINEAR CIRCUIT DESIGN
Mike & Richard Tooley

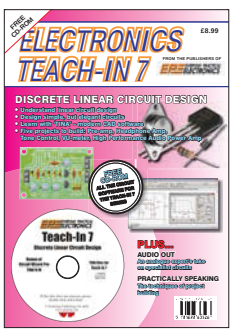
Teach-In 7 is a complete introduction to the design of analogue electronic circuits. It is ideal for everyone interested in electronics as a hobby and for those studying technology at schools and colleges. The CD-ROM also contains all the circuit software for the course, plus demo CAD software for use with the *Teach-In* series.

- Discrete Linear Circuit Design
- Understand linear circuit design
- Learn with 'TINA' – modern CAD software
- Design simple, but elegant circuits
- Five projects to build:
  - i) Pre-amp
  - ii) Headphone Amp
  - iii) Tone Control
  - iv) VU-meter
  - v) High Performance Audio Power Amp.

PLUS

Audio Out – an analogue expert's take on specialist circuits

Practically Speaking – the techniques of project building.



ELECTRONICS TEACH-IN 8 – CD-ROM
INTRODUCING THE ARDUINO
Mike & Richard Tooley

Hardware: learn about components and circuits
Programming: powerful integrated development system
Microcontrollers: understand control operations
Communications: connect to PCs and other Arduinos.

Teach-In 8 is an exciting series designed for electronics enthusiasts who want to get to grips with the inexpensive, popular Arduino microcontroller, as well as coding enthusiasts who want to explore hardware and interfacing. It will provide a one-stop source of ideas and practical information.

The Arduino offers a truly effective platform for developing a huge variety of projects; from operating a set of Christmas tree lights to remotely controlling a robotic vehicle through wireless or the Internet.

Teach-In 8 is based around a series of practical projects with plenty of information for customisation.

This book also includes *PIC n' Mix*: 'PICs and the PICkit 3 – A Beginners guide' by Mike O'Keefe and *Circuit Surgery* by Ian Bell – 'State Machines part 1 and 2'.

ORDER YOUR BUNDLE TODAY at www.electronpublishing.com

AOShop

The home for specialist audio, analogue and historic components – provided by Jake Rothman, *PE's Audio Out* columnist. The AOShop is your best bet for classic analogue 'NOS' (new old stock) components, including all parts for *Audio Out* projects and designs.

Ge semiconductors

Small-signal PNP transistors
NKT214F, OC57, OC59, 2N1377, 2N525 **£1.00**

Low-noise PNP transistors
GET106 **£2.50**

Small power PNP transistors
AC153, AC153K, AC188K **£1.50**

Small power NPN transistors
AC176, AC141K **£2.00**

High-voltage PNP transistors
OC77, CV7001 **£1.50**

TO3 PNP power transistors
AD149, AD161, AD162 **£2.00**
AD140 **£3.50**
AD149, AD161/2 matched pair **£5.00**

RF PNP transistors
OC41, OC42, 2SA12, 2SA53, AF124, AF178, GET872A **£1.50**

NPN transistors
OC139, OC140, ASY73 **£2.00**

Diodes
CV7049 (OA10), CG92 (OA91) **£0.50**

Si semiconductors

Diodes
ZC5800 RF Schottky **£0.20**

Low-noise silicon transistors
BFW16A, 2SC3071, 2SC3068, 2SA1016K, 2SC2362K, 2SA970BL **£1.50**
2SC2204, 2SD655, BC550C **£0.50**
ZTX651 **£0.30**
ZTX751 **£0.50**

RF transistors (suitable for Theremin)
BF180 **£0.40**
BF199 **£0.50**

Audio power MOSFETs
Exicon 10N20, 10P20 **£6.50**
Hitachi 2SJ99, 2SK343 **£3.50**
Hitachi 2SJ56, 2SK176 **£8.50**

JFETs
BFW11, BFW10, TIS73L, J177, J113, U1994, U1898, 2SJ176, J201 **£1.00**
J175, J176, J112, J111, 2N3820, 2N5457, BF244, 2N5460, J230 **£0.60**

Small power output/driver transistors
2SB649A, D669A, 2SA1208, 2SC2910, MJE253G, MJE243G, 2SA1725, 2SC4511 **£1.20**
BD139,BD140, BD135, BD136, BD435, BD436, MPSA63, BCV46 **£0.50**

MOSFETs
ZVP2106A **£0.30**
Dual-gate 3SK45, BFS28 **£1.50**

Metal-cased transistors
BCY71 **£0.30**

Amplifiers
LM318 high-speed op amp **£0.35**
µA709 metal-cased op amp **£2.00**
5534H metal-cased op amp **£3.00**
LM384, TDA2030A, TDA2050V **£1.50**
TL082 (ceramic pack) **£1.00**
TAA435 (Mullard power amp driver for AD161/AD162) **£1.20**

ICs

That/dbx 2180 VCA/VCF **£6.00**
CA3080 VCA/VCF (vocoder) **£2.80**
CA3280 VCA/VCF **£4.00**
LM13600/ LM13700 VCA/VCF **£2.00**
PT2399 echo/delay **£1.50**
Original TI 44V NE5532 **75p**
Nisshimbo 44V NE5532DD **£1.00**
SMT version **50p**
New 36V TI NE5532 **50p**
Original TI NE5534A **£1.40**

Dual transistors

2N2639, 2N2223, 2N2910 (NPN) **£4.00**
E401 (JFET Moog) **£4.00**
2SK2145-Y dual JFET **£0.80**
2N5564 JFET **£8.00**
HN3C51F, HN3A51F **£1.00**
DMMT3904/6, HN1A01F, HN1C01F **£0.50**

Loudspeakers

PE Mini-Monitor
Volt PE165 6.5-inch woofer (each) **£85**
Morel MDT29 tweeter (each) **£25**
Kit *pair* of PE165/MDT29 plus Volt crossover parts and PCBs **£299**
Monacor DT-28N tweeter (each) **£35**
Vifa 19mm BC20SC15-04 tweeter (each) **£15**

Volt crossover inductors
1.2mH, 1.5mH, 2mH, 2.7mH, 0.5mH (tapped at 0.3mH) (each) **£5.85**

Fully tested reclaimed speakers
Vifa BC14 5-inch woofer (each) **£10**
Vifa TC26 1-inch tweeter (each) **£10**

Low-price speakers
Philips 4-inch 4070 **£2.00**
EMI 10x6-inch, 30Ω Alnico **£7**
64mm 64Ω neodymium **£1.20**

Capacitors

Note '10/63' denotes '10µF 63V'.

Polyester
3.3/100, 4.7/250, 4.7/63 **£1.00**
5.6/63, 8.2/63, 10/63 **£2.00**
Mullard 'Mustard' C296 0.22/400 **£2.00**

Polycarbonate
Axial 2.2/63 1%, 4.7/160, 6.8/63 **£1.00**

Radial
6.8/160V, 10/63 **£2.00**
Reclaimed 22/63 **£2.00**

Polystyrene
Philips 1% 4.7nF/160, 6.2nF/500, 12nF/63, 22nF/63, 110nF/63, 24nF, 2nF **£1.00**
RIFA 2% 100nF for active xov. **£1.00**
Suflex 90.9nF 0.5% **£2.00**
Suflex 2.5% 10nF/63 (rad. or ax.) **£0.50**

Radiation resistant
Siemens cellulose acetate MKL 2.2/25 **£0.80**

Electrolytic – Mullard blue 017 series
10/25, 22/25, 100/10 **£0.50**
150/40, 470/40, 1000/25 **£1.00**

Tantalum – axial metal cased
22/50, 47/35, 68/25, 100/20, 120/10, 150/16, 220/10, 330/6 **£2.00**
22/35, 33/35, 47/20, 68/15, 100/10, 150/6 **£1.25**
4.7/50, 6.8/35, 10/25, 10/35, 22/15 **£1.00**

Axial moulded-case tantalum
Kemet axial 6.8/10 **£0.30**
Kemet radial 33/10 **£0.40**
STC radial 100/20 **£1.50**

Tantalum bead
22/50 **£1.00**
470/3 **£2.00**
680/6.3 **£3.00**

Wet tantalum
220/25 axial **£3.20**
Castanet button 140/30, 470/3 **£3.20**
Hughes 540/10 **£3.20**

Bipolar
Hermetic bipolar tantalum 16/35 **£3.20**
Elcap axial 10/50 **£0.50**
Generic radial 100/16, 470/35, 100/ 63, 22/35, 4.7/35, 220/16 **£0.50**

Philips solid-aluminium (axial) 121/123
47/16 **£1.00**
330/6.3 **£2.00**
100/35 **£4.00**

Philips Pearl 122 series (radial)
10/16 **£0.30**

Silvered mica
1nF/125 axial 1% **£0.30**

Trimmer capacitors
Vishay plastic-film 4-40pF **£0.80**
Vishay plastic-film 5.5-45pF **£1.00**
Vishay plastic-film 5-80pF **£1.20**
Vishay plastic-film 10-250pF **£2.00**
Mica 1-12pF, 2-40pF **£1.00**

Audio transformers and inductors

Eagle transformers
LT44, LT722 driver, LT700, LT723
500Ω output **£2.50**
LT30 500mW output **£3.50**
5:1 interstage (fuzz box) **£1.50**

Reparco
T/T3 splitter transformer **£4.00**
CH2 5mH RFC **£2.00**

Balanced output transformer
Vigortronix 600Ω VTX-101-007 **£10**
Vigortronix 600Ω VTX-101-3001 **£10**
Vigortronix 600Ω VTX-101-3002 **£15**
Gardners 150Ω **£10**
Reclaimed BBC LL74/MPC nickel core 600Ω **£12**
Low-level 1:1 600Ω (modem) **£2**
Reclaimed mic input transformer **£15**

Inductors
82µH, 4.7mH, 100µH, 270µH, 10µH, 14µH (low Z) **£0.50**
4-inch ferrite rod with MW and LW windings **£3.00**

Special resistors

Bourns wire-wound trimmer
5kΩ 3059 JM panel-mount **£2.00**
Thermistor RA53, R13 **£4.00**
A13 **£2.00**
Thermistor CZ1, CZ6 **£1.50**
Holco H2 2.2MΩ 1W, 1% **£1.00**
Welwyn 1GΩ 2W **£1.00**
5k Bourns 3321H cermet trimmer **£0.50**

Potentiometers

Bourns 81
25kΩ lin cermet **£2.00**
25kΩ lin, 5kΩ lin cond. plastic **£3.00**

Bourns 91
10k dual-gang lin or log **£10.00**

Plessey moulded-track
5kΩ log with switch **£3.00**
50kΩ A/log **£3.00**
Mil 250kΩ lin dual **£5.00**

Bourns
8-gang 50k A/log **£7.50**
ALPs 4-gang 100k A/log **£3.00**

Alpha
16mm 4.7kΩ A/log **£0.80**
220kΩ A/log **£0.80**
10kΩ lin centre-detent dual-gang **£1.50**

Allen Bradley J series/Honeywell
10kΩ lin 1 million cycles **£5.00**
Blore Edwards AB 45 dual 5kΩ A/log with switch **£3.50**
Alps RK9 dual-gang 5kΩ RD law **£4.00**
Alps 50k log tapped motor. stereo **£5.00**
BI P260 500kΩ log conductive plastic 1 million cycles **£2.00**
Colvern wire-wound 100kΩ or 50kΩ dual-gang 3W **£5.00**
Mil Spec hermetic 10Ω **£8.00**

Miscellaneous

PCBs
Phase-linear crossover PCB **£3.00**
8-gang All-pass Filter PCB **£3.00**
8-gang Pot PCB **£1.50**
Polarity Inversion PCB (Alps) **£1.50**
Polarity Inversion PCB (alt.) **£1.50**
All-pass Filter PCBs (three) **£4.00**
.. Rotary Switch (AO0626ROT) **£1.50**
.. Pot Board (AO0626POT) **£1.50**
Silent Stomp Switch (pair) **£4.00**
Penta-fuzz Bender (*PE*, Feb26) **£3.00**
Pre-driver (*PE*, Jan26) **£3.00**
Pre-driver jack bd (*PE*, Jan26) **£1.50**
Fuzz Box (*PE*, Nov25) **£3.00**
Fuzz Box jack bd (*PE*, Dec25) **£1.50**
Dual JFET Buffer (*PE*, Oct25) **£2.00**
Mosfet Buffer (*PE*, Oct 2025) **£2.00**
Dual Mosfet Buffer (*PE*, Oct25) **£2.00**
Cascode Amp/PS (Sep 2025) **£2.00**
Pocket Theremin (*EPE*, 1996) **£2.00**
Elysian Theremin (*EPE*, 1996) **£6.00**
Synth VCF, VCO (*EPE*, 2017) **£3.00**
48V PSU (*EPE*, 2019) **£3.00**
SMT dual transistor adapter **£0.40**

Switches Blore Edwards/Plessey/etc
ALPs 4-pole push switch **£1.50**
MH1 4-pole 3-way **£3.00**
Elma 01 2-pole 6-way **£6.00**
3PDT guitar pedal stomp switch **£3.50**

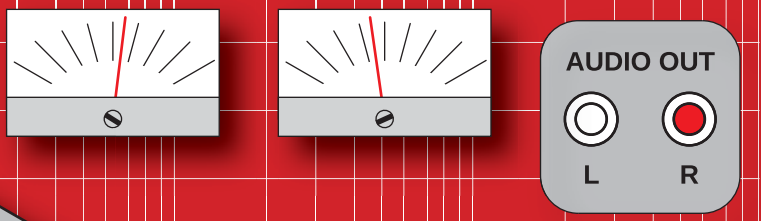
Valves
EC88, PCL84 **£3.00**

Contact
Jake Rothman
Manchester House
Market Street, Craven Arms
Shropshire SY7 9NN
(visit by appointment)
+44 (0)1597 829102
jrothman1962@gmail.com

Payment
PayPal, bank transfer, cheques (payable to "J Rothman", UK pounds only). No VAT payable

Postage
Small Jiffy bag £2.99
Small package £4.99

AUDIO OUT



By Jake Rothman

The RetroTech UK event and radio rallies, part 1/2

For the last three years, a vital date in my calendar has been RetroTech UK, an event run by the British Vintage Wireless Society (BVWS – bvws.org.uk). It's a wonderful sale of old electronic stuff that takes place every May.

It gets better each year as more people realise that modern gear, with its proprietary subscription-based software, glued-together plastic cases and irreparable surface mount PCBs with no schematics offer little for the electronics enthusiast.

It is amazing what comes out of people's lofts and workshops: valve amplifiers, top-notch test gear, analogue (surely we have to use the old British spelling here) telephones as well as old radios and components.

Photo 1: a 405-line TV from the 1950s. It could be hired out as a prop for movies and theatre.

Photo 2: a close-up of the 405-line picture.

There are also numerous radio rallies in the warmer months. These events are normally held in odd locations such as airfields and leisure centres, so a car is required.

A few hundred quid is also needed, since most traders are septuagenarian, cash-only guys. If you meet one who says "it's worth £xxx on eBay" walk away until he sees sense at the end of the show. If not, watch with schadenfreude as he humps it back into his car.

Click to brick

When you buy old technology, you own it, unlike the new stuff where you are tied to a company or server to generate recurring revenue. Therefore, it can't be 'bricked'. Bricking is when a software service is stopped, rendering a piece of electronic hardware into a useless, non-functioning lump, such as the Robert's DAB radio debacle a few years ago.

Occasionally, analogue systems are discontinued, as with the 405-line black-and-white television service when the last transmitter was switched off in 1985. Millions of TVs, such as the one shown in Photo 1, became 'bricks' overnight.

I knew someone who made theirs into an aquarium, but now you can get a 405-line converter in a small box rather than a five-foot rack. This means old sets can be brought to life again, as shown in Photo 2. The picture quality really was that bad, but we accepted it until 625 lines and colour came along in 1968. I knew an old lady who used one until 1982.

CRT TVs became less useful with the switch from analogue to terrestrial digital TV, then again from cathode ray tubes to LCD screens and now from transmitters to internet streaming.

Longwave lament

The latest 'bricking' is the BBC's decision to switch off the 198kHz 500kW longwave transmitter at Droitwich on June 27th. It had been running since 1934. It was the last service that could cover the whole country and beyond. It could be received by simple circuits, such as crystal sets and old AM radios, which can run on a small battery.

The present Radio 4 service used to be called the Home Service up until 1967. The country's civil defence messaging system is now dependent on smartphones and a secure power supply from the National Grid. Our adversaries could knock out both in a few minutes. Although whether

Photo 3: because of the AM shutdown, old radios are cheap. It's much easier to feed an audio signal into a radio than a video signal into a 405-line TV.

there would be anything useful to transmit, given the present civil service malaise, is debatable.

The Shipping Forecast, a British icon that once inspired me to write a song of that name, will also go (although it may continue on satellite and FM). The reason cited for this crass decision is the difficulty of maintaining the old valve technology. I suspect the reality is the transmitter's continuous electricity consumption of 700kW (DAB transmitters use less power but the receivers use more) and the valuable real estate occupied by the huge antennae.

With the UK having the most expensive electricity in the industrial world, and the BBC's declining licence fee revenue as it commits suicide by becoming just another streaming service, it all seems sadly inevitable. Despite the BBC's academic bias, I will deeply miss their science programmes and listening to *The World Tonight* in the bath.

Since I can't get Radio 4 on DAB or FM here in Craven Arms, I may have to rig up a low-power longwave transmitter fed by computer, streaming it online, to continue using my wonderful Bush TR82. I could also sacrilegiously stick a Bluetooth receiver module in it, but they consume over twice the current the radio does. I certainly don't want a 'smart speaker' in the bathroom.

The good news is that the prices of early germanium transistor radios, like those in Photo 3, have tanked. I bought three broken Bush TR82Bs for £3.00 each; they used to command £25. The same applies to low-bandwidth (20MHz) CRT oscilloscopes. I picked up two Hameg HM203-6 CROs for £5.00 each and they were working! (See Photo 4.)

I also got a VTR103, a version of the TR82 with a VHF band, which will still be useful if I rig up a proper FM aerial. *Silicon Chip* published articles on the TR82C in September 2013 and the VTR103 in August 2021, both by Ian Batty. It would make a good subject for *Audio Out* to have another go at restoring and repurposing these radios.

Even if they are useless as radios, they are worth much more than their selling price for their OC series Mullard transistors and chunky audio transformers (see Photo 5), useful for electric guitars, effects and studio colouring units.

Later Bush radios, such as the TR82C and the VTR103, have AF117/4 series transistors, which suffer from tin whiskers in their metal cans, so expensive replacements

Photo 4: the age of the £5.00 scope has arrived; catch it while you can!

Photo 5: the chunky AF transformers and OC81 transistors in old transistor radios are highly sought after, such as these from a Bush TR82.

Photo 6: old telephones are well catered for at RetroTech.

are often needed. All electronic workshops should retain a longwave radio anyway, for lightning/arc detection and low-frequency electromagnetic compatibility (EMC) sniffing. It's essential kit for audio switch-mode circuitry design.

Telephony

The UK analogue telephone network is almost gone now, as it too migrates online. Still, it's liberating £1.5 billion in copper scrap for Openreach as it's replaced by fibre. Again, resilience is being sacrificed for what? Short-term shareholder gains?

Still, for us minions who tinker in our workshops, there is a network of people who cherish the old telephones, the technology that started the whole audio business. They were out in force at RetroTech (see Photo 6).

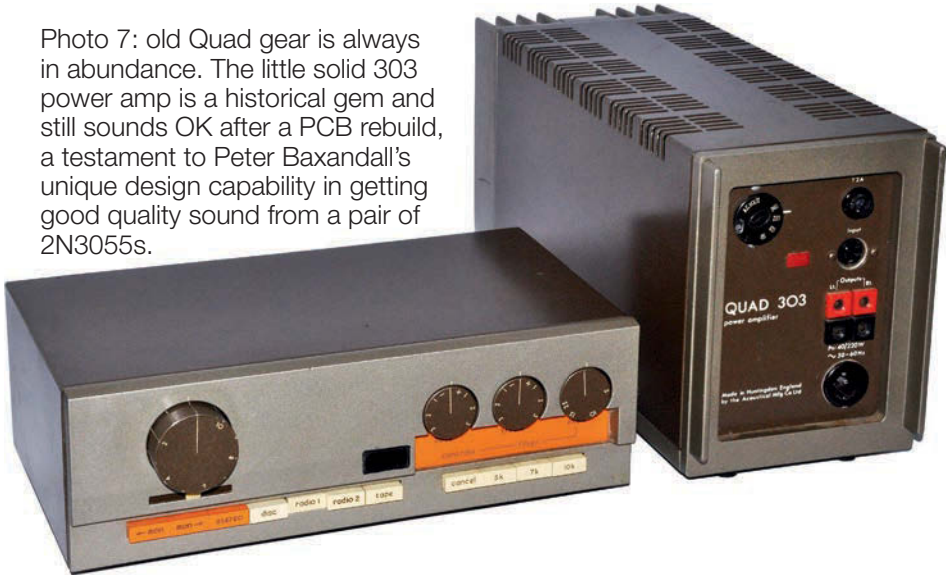


Photo 7: old Quad gear is always in abundance. The little solid 303 power amp is a historical gem and still sounds OK after a PCB rebuild, a testament to Peter Baxandall's unique design capability in getting good quality sound from a pair of 2N3055s.

Some people wire them up between their houses. They even had the red 21A carbon microphone dynamic replacement microphone inserts. I remember upgrading our microphones in the late 1970s.

Did you know that there's even a telegraph pole appreciation society?

Infrastructure decay

I regard this ongoing destruction of analogue infrastructure as a big mistake, similar to ripping up half of Britain's railway network for scrap between the 1960s and early 1980s. The medium-wave (AM) transmitters are going next year. All this was designated critical national infrastructure, but it is those who own it who are destroying it, so we can't do anything.

I signed a petition to retain LW, but it only got 8000 signatures. Rajar, the radio joint audience research agency, gave up even measuring it.

Photo 8: I was surprised to meet radio rally regular Tony "Smith" (his real Polish name I couldn't transcribe) at the show. I get my soldering gear from him. He's still got lots of proper leaded solder.

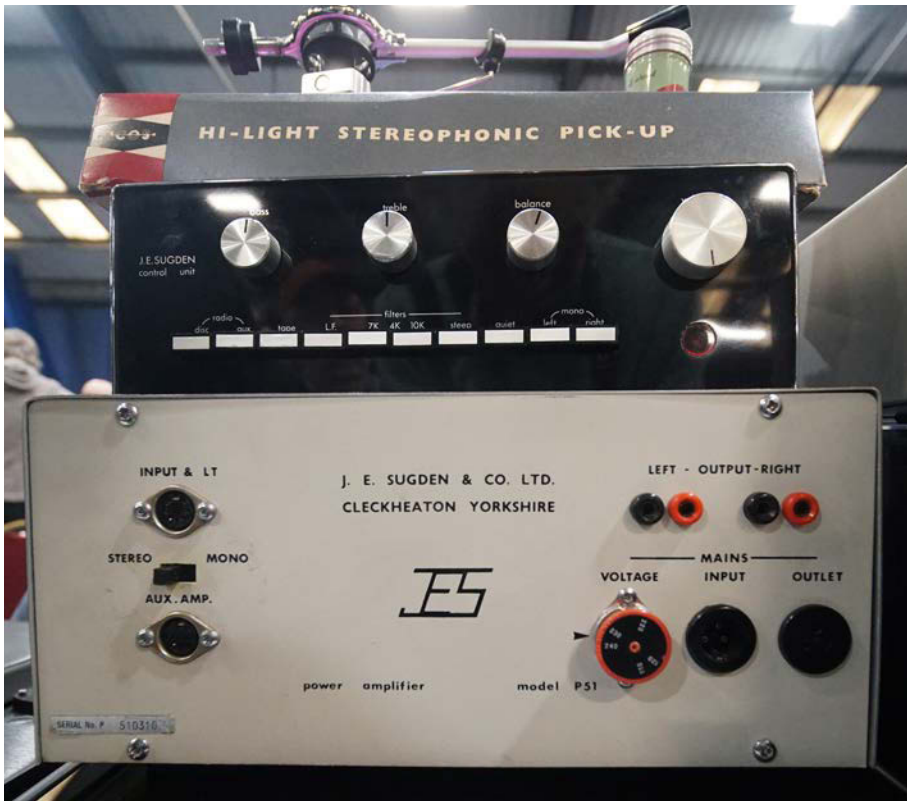


Photo 9: another early transistor amp that sounded good because of its simple Class-A circuit. Only 10W, though.

Photo 10: I always liked the unique appearance of Quad preamps, especially the filament-lamp-illuminated logo.

Does it work?

Don't bother asking; "no it doesn't" is usually the answer. Most of the gear at these shows has been fiddled with by enthusiastic amateurs. Expect to do a lot of work to bring things up to scratch.

Still, that's the fun of it! You can't complain if you bought a radio for less than the price of the batteries. If you want restored collectible stuff, go to Audio Jumble in Kent, but expect to pay eBay "sold-for" prices, which will be five times more than at the rallies.

Amplifiers

Everything in the history of audio amplifiers is represented at RetroTech, from early 1920s triode amplifiers, wired with screw terminals on mahogany chassis, to Class-D Chinese modules. I generally stick to buying discrete transistor designs.

One that always turns up is the Quad 303 power amp, introduced in 1967 and designed by Peter Baxandall (Photo 7). It was the first transistor amplifier to sound as good as valve designs. In fact, this year there was a whole stand devoted to Quad, with lots of spare parts. I nearly bought a Quad 44 preamp (the only Quad preamp that's any good) for £120. I dithered, came back, and it was gone.

You can't dither at these shows. Just by picking stuff up and looking at it, you generate interest from

Photo 11: a very expensive, top-quality Hi-Fi amp from *Practical Wireless*. Unlike many home-constructed kits, this one was built to professional standards.

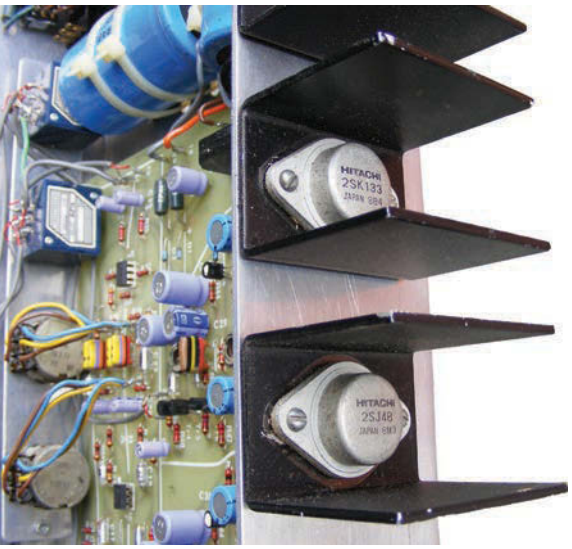


Photo 12: the Winton used very reliable and robust metal-cased Hitachi lateral Mosfets.

Photo 13: a Mullard 5-10, a historical audio milestone. Silly me, I didn't buy it because it was mono.

others. Take a gamble, buy it, and put it back in next year's show if you don't like it (after you've fixed it, photographed it and written an article on it).

I used to help out a radio rally guy called Colin Rich; I frantically rummaged through the front of his stall, which always generated extra sales. Some of the radio rally traders have stands at RetroTech, shown in Photo 8.

Photo 9 shows one interesting early transistor design, the Sugden Class-A. This avoided the Class-AB crossover distortion that typified many transistor designs at the time. It's very similar to the famous John Linsley-Hood design (he wrote for *PE*) but gave about 10W of output.

Older preamps such as the Quad 33 (see Photo 10), or others with lots of interlocking pushbutton switches, are usually an intermittent contact disaster. The sound of this one was not so good as it was very noisy due to the use of old carbon composition resistors and EF86 pentodes.

Seller's regret

Often one has to buy things again that one already had, but then later sold, regrettably. Photo 11 shows the *Practical Wireless* Winton amplifier. It was £25 at the start of the show, but still there at the end, so I cheekily offered £15. I had built one from a kit for one of my father's friends in 1980 and found it so good that I wish I had kept it.

It used the then-fashionable Hitachi lateral Mosfets (Photo 12). Time has since shown these TO-3 package devices to be the most reliable audio output devices ever made, and in this example, they were still working as expected. Only now have they stopped being made (by Exicon).

They fell out of fashion because they were inefficient, having an on-resistance of 1.2Ω, needing a 20% bigger mains transformer for the same power as bipolar transistors. They also ended up being six times the price. With high-impedance (>8Ω) loudspeakers, the generic Hitachi application circuit (used here) works very well.

With the advent of 4Ω loudspeakers, where putting devices in parallel was necessary, their fate was sealed. When I later found it had been fitted with Alps Blue Velvet pots as well, those and the Mosfets alone more than paid for the petrol and entrance fee.

Photo 13 shows a Mullard 5-10 amplifier. Oh, why didn't I buy it? What an idiot I am! The Sowter output transformer alone costs £200. The reason I didn't was that he only had the one and I was fixated on stereo.

I forgot about the Blumlein's original middle and side two-channel system, where the two channels do not have to be matched. This means you can have a good stereo image with completely different speakers and amplifiers. I bet it won't be there next time.

Speakers

I've long been a loudspeaker lover and RetroTech always provides lots of goodies. The rallies are only good for radio loudspeakers, not Hi-Fi. One advantage of loudspeakers is that the technology advances relatively slowly, so a good late-1970s pair of speakers can still sound very respectable today, especially if they have been well looked after or restored.

It's only if you want loud volumes that older speakers can be a bit disappointing.

The only time I was ever fooled into thinking reproduced sound was real was way back in 1984. It was from a pair of Quad ELS63 electrostatics

driven by a Philips CD player and a Quad 405 amplifier. I really thought someone was playing a piano in the living room. Modern loudspeakers are more coloured since they abandoned bituminous cabinet damping, cone doping and complex crossovers; I've never been fooled since.

Photo 14 shows some of the earlier Quad electrostatics, the ELS57. They sound very clean but are too directional, making them sound distinctively 'loudspeakerish'. Fellow LS3/5A enthusiasts were well catered to by three stands selling complete systems and drive units (Photo 15).

I even bought a stack of open-circuit KEF T27 tweeters for £25, shown in Photo 16, because I'm used to back-winding the voice coils. New ones cost £125 each at Falcon Acoustics. Crossovers often have difficult-to-obtain components. I bought a pair of three-way third-order MA83 crossovers, shown in Photo 17, for a couple of quid. That wouldn't even buy you one new inductor.

Finally, another weird loudspeaker thing, a Murphy open-baffle radio (Photo 18). Murphy always used to strive to get good quality sound out of their designs before the phrase Hi-Fi had been thought of.

Old BBC gear often turns up, and the LS5/8 active speaker system shown in Photo 19 was a fantastic piece of audio history. It was a two-way system with a 300mm Harbeth-designed polypropylene cone combined with a 34mm Audax tweeter. The crossover was a low 1.7kHz and the circuit is in the BBC report BBC RD 1979/22. It was the BBC's last and biggest monitor design.

Photo 14: this stand is always there, selling ex-BBC stuff. There were Quad Electrostatics and BBC 1960s silver monitors; it's where I got the PPM meters.

Photo 15: this man sold home-made LS3/5a speakers for £300-400 pair using recycled KEF B110s and T27s in 2025. I have friendly competition!

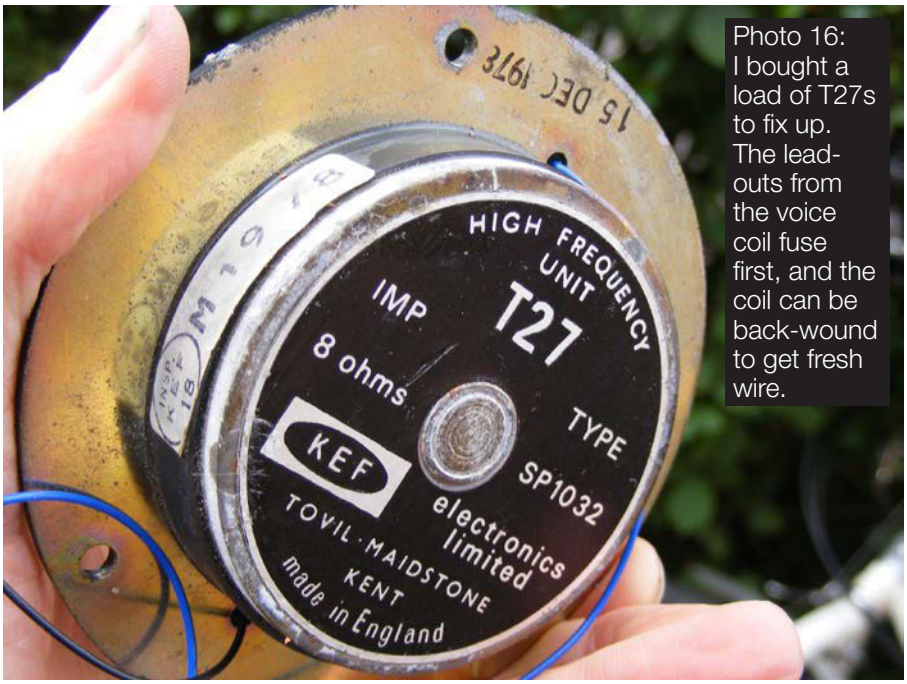


Photo 16: I bought a load of T27s to fix up. The lead-outs from the voice coil fuse first, and the coil can be back-wound to get fresh wire.

Videotech

As well as all the trade stands where you could buy an old TV, there were exhibitions by the Broadcast Television Technology Trust (www.bttt.org.uk) and the Broadcast Engineering Conservation Group (https://becg.tv), who run the Broadcast Engineering Museum in Lincolnshire. Note that they are having an open day on the 2nd of August.

There were lots of old CRT TVs showing real pictures (Photo 20) and even an old Baird mechanical 30-line Televisor!

TV cameras with tubes

When I was a nine-year-old, back in 1971, I was fascinated by the huge TV cameras that were used to film the live Blue Peter show. They said EMI

Photo 17: passive crossovers are very cheap at RetroTech. One guy who sold Stirling Audio LS3/5as even gave me a pair from transmission-line R50s.

Photo 18: these open-baffle Murphy radios had better sound than the competition at the time and were almost a piece of walnut furniture. They were made from around 1934 (A24) to 1949 (Batwing 188). This is a later model.

Photo 20: I found the first decent TV my dad rented in 1970, a Murphy with an 8×5-inch elliptical speaker. You could get them in crazy 1960s colours, like white, teal and orange, which we had. I remember the TV repairman coming round and swapping valves. He got cross because I tried to help!

Photo 21: a classic Marconi TV camera, the type that would have been used for the Queen's coronation in 1953. My mum remembers watching it at the only house with a telly down the road in Muswell Hill, London.

Photo 19: an early BBC active speaker system built into a Quad 405 stereo power amp. It should be in an audio history museum. I don't think there is such a museum in the UK, though (yet).

on the side and had zoom lenses eight inches (about 20cm) in diameter.

I was amazed to see an operational earlier Mk.3 camera designed by Marconi in 1953 at RetroTech (Photo 21). It used an Image Orthicon tube for the sensor, basically a cross between a photomultiplier and a cathode ray tube in reverse with a scanned grid. They were used before the introduction of the charge-coupled device (CCD) sensors (in 1980) that are common today.

Next month

We'll conclude this report next month with some more good old analogue technology.

P. S.: I'm going to the Telford radio rally on the 6th of September. PE

By Andrew Levido

Power Electronics

Part 6: DC-to-AC Converters

Having covered DC-DC and AC-DC converters in earlier articles in this series, we will now move on to DC-AC converters. They have been around for a long time, but their usage has become widespread over the last couple of decades.

There was a time when the applications for DC-to-AC converters were limited to industrial motor drives and commercial UPS systems. However, the widespread adoption of domestic solar power systems and electric/hybrid vehicles means that DC-AC converters have become extremely common.

As has become the custom for this series, we will start by analysing the simplest possible converter and build from there. Fig.1 shows four switches arranged in a H-bridge, fed from a DC voltage source. If we close S1 and S4 for a period equal to a half-cycle of the desired output frequency and S2 and S3 for the other half cycle, we can synthesise a square wave with amplitude V_{SRC} and frequency ω, shown in red.

Recall that ω (omega) is just a frequency expressed in radians per second; 2π radians is one cycle, so 2π radians per second is the same as 1Hz.

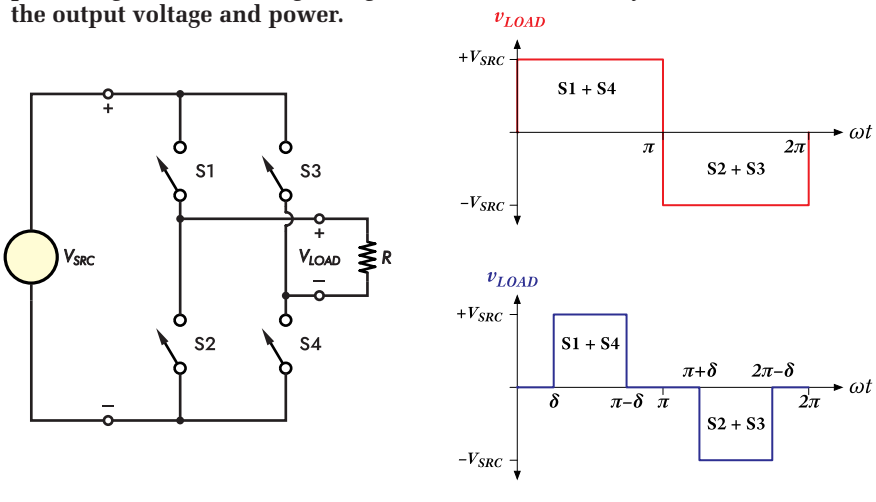
There are a couple of important things to note about this circuit. First, energy passes from the DC side to the AC side, so this is an inverter; in a rectifier, it goes the other way. Secondly, the switches create an AC voltage source at the output terminals, so this is a voltage-source inverter. The need for these seemingly obvious observations will become apparent soon.

The RMS value of the output voltage is just V_{SRC}, so the only way to change v_{LOAD}, or the power delivered to the load, is to adjust V_{SRC}.

We can change this by introducing a third switching state into the cycle, one where all switches are off and the output is zero, as shown in the blue trace. Here, the switches are off for a phase angle of δ (delta) at the beginning and end of each half-cycle. We can now use δ to adjust the RMS output voltage, and hence the power delivered to the load, according to the expression $v_{LOAD(RMS)} = V_{SRC}\sqrt{1-2\delta \div \pi}$.

This kind of inverter is sometimes known as a tri-state inverter because each output terminal can be connected to $+V_{SRC}$, $-V_{SRC}$ or zero (another term you might see referring to similar systems is 'modified sinewave inverter').

Fig.1: the simplest single-phase voltage-source DC-AC converter uses four switches to create a square wave output. By leaving all switches open for a phase angle of δ at the beginning and end of each half-cycle, we can control the output voltage and power.



Most of the interesting loads we want to drive are not purely resistive, so we should see what happens if we add an inductor to the load (Fig.2). We will assume that the L/R time constant is large compared to the output frequency so that the AC current can be approximated by its fundamental component.

There is a phase shift between the voltage and current of Ø radians, as we would expect with an inductive load. This angle is fixed by the relationship between the load inductance and resistance, so it does not help us to change the power delivered to the load.

Things are different if the load includes an AC voltage source, as shown in Fig.3. This scenario is pretty common; for example, in a solar inverter feeding the power grid or an inverter driving a synchronous motor with its sinusoidal back-EMF.

This circuit gives us another variable to play with, as we can control the phase angle between v_{LOAD} and v_{AC}. With this circuit, you can even dictate the direction of power flow – from source to load (inversion) or load to source (rectification).

We effectively have two AC sources: the inverter output v_{LOAD}, and the AC source, v_{AC}, with an inductor between them. The inductor acts as a 'buffer' between the two, absorbing the instantaneous voltage difference. The greater this difference is, and the longer it has to be sustained, the larger the inductor required.

Still, a large inductor reduces the power factor, limiting the real power that can be transferred.

The solution is to move the inductor to the DC side of the H-bridge, where it won't impact the power factor, no matter how large it is. This arrangement (Fig.4) means that the output of the H-bridge is an AC current, so this is now a current-source inverter. The

DC voltage source and inductor are equivalent to the current source I_{SRC}.

We can't use the same switching strategy to achieve the 'zero-state' output as we did in the voltage-source inverter. If we turned all the switches off, the source current would have nowhere to go and the voltage across the H-bridge would rise uncontrollably. Instead, we achieve the zero state by turning both S1 and S2 or S3 and S4 on at the same time, diverting the current away from the load.

I have shown the current-source inverter switching pattern in the figure.

This 'shoot-through' would be catastrophic in a voltage-sourced inverter, as would a short on the output, because the source would be short-circuited and the resulting current uncontrolled. Current-source inverters are inherently current-limited, making them very robust compared to their voltage-source counterparts, which require additional circuitry to protect against short circuits.

I won't go through the maths, but it is possible to show that the power delivered to the load by this current-source converter is given by the equation $P = (2v_{AC} I_{SRC} \div \pi)\cos(\delta)\cos(\emptyset)$. We can control the power flowing to or from the load by changing either δ (the time spent in the zero state) or Ø (the phase angle between the inverter output and v_{AC}), or both.

These are dictated by switch timing, so they can be nicely varied by a microcontroller-based implementation.

Full control of Ø requires true bidirectional switches – switches that can conduct or block in both directions. This is somewhat difficult to implement, since Mosfets and IGBTs both normally have an anti-parallel diode that means they can never block reverse current.

In the case of Mosfets, this is the inherent body diode, and in the case of IGBTs, it is usually a separate diode integrated into the package. Typical applications therefore tend to leave Ø at or very near zero and manipulate δ to control power.

We have used single-phase DC-AC converters so far, but everything we have covered is equally applicable to three-phase converters. Fig.5 shows that these are just a single-phase converter with an extra pair of switches. Each phase-to-phase voltage is created by manipulating the switches

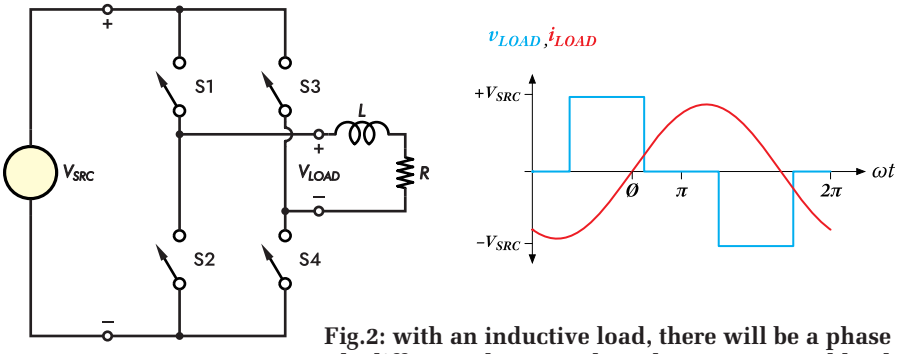


Fig.2: with an inductive load, there will be a phase angle difference between the voltage output and load current that is determined by the load inductance and resistance.

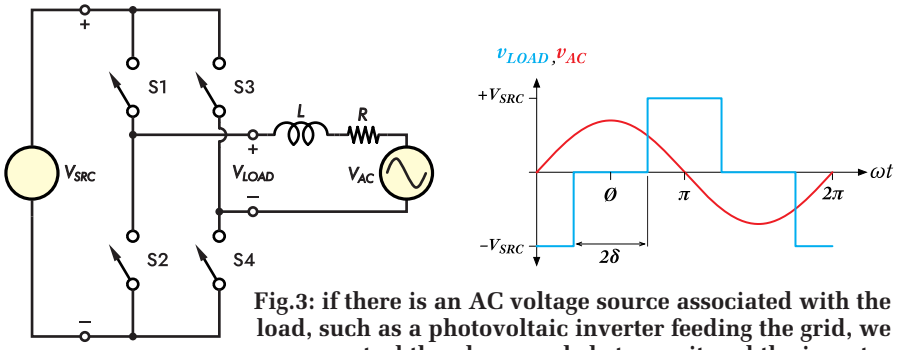


Fig.3: if there is an AC voltage source associated with the load, such as a photovoltaic inverter feeding the grid, we can control the phase angle between it and the inverter output, allowing us to control the power flow in either direction.

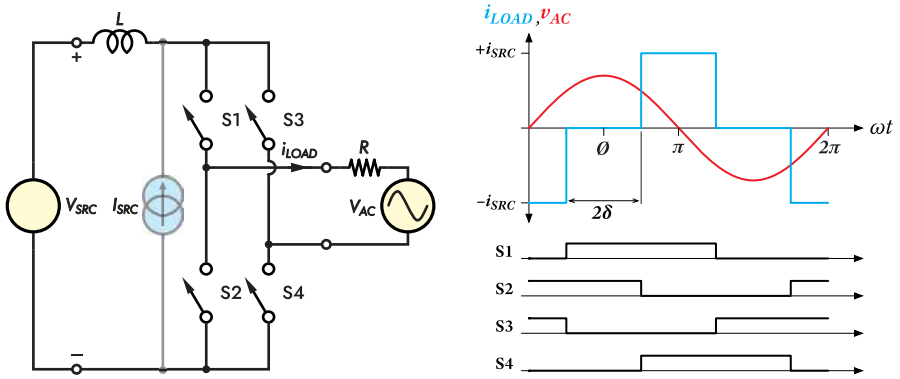


Fig.4: if the inductor in Fig.3 is moved to the DC side, it results in a current-source inverter. The zero-state output occurs when switches S1 and S2 or S3 and S4 are on.

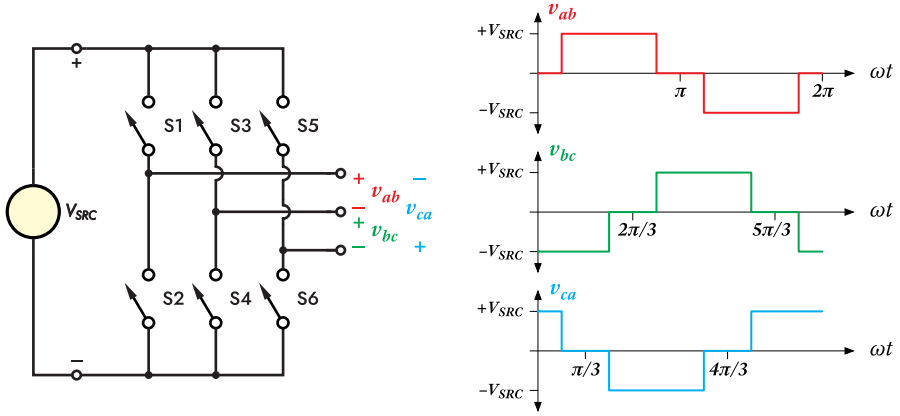


Fig.5: a three-phase DC-AC converter is just like its single-phase counterpart with an extra pair of switches. The phase shift between switching is 2π/3 radians (120°) instead of the π radians (180°) in the single-phase case.

just like the single-phase example, but shifted by 2π/3 radians (120°) instead of π radians (180°) in the single-phase case.

You could, in theory, build an n-phase converter with n switch pairs (or two pairs for single-phase) with a phase shift of 2π ÷ n between them. However, diminishing returns means we rarely deal with more than three phases in reality.

The exception is primarily in aircraft and ship motor drive systems, plus some large-scale mining equipment, where more phases (eg, six or 12) can result in lower torque ripple, a lower current per phase, reduced harmonics and EMI, plus better fault tolerance.

Pulse-width modulation

The examples we have seen so far have all produced more-or-less square-wave outputs, which we know have a terrible power factor, so are not very efficient at transferring power. Practical DC-AC converters therefore employ some kind of modulation scheme to reduce the level of harmonics in the output, or at least push them up to high frequencies where they are easier to filter.

One common method used for small to medium power converters (up to a few hundred kilowatts) is sinusoidal pulse-width modulation. The switch drive is modulated at a carrier frequency (f_{PWM}) much higher than the desired output frequency.

The on-time in each modulation period is proportional to the instantaneous amplitude of the desired sinusoidal output waveform, as in Fig.6(a). I have shown only one half-cycle for clarity; the other is identical but reflected in the horizontal axis.

In that diagram, the carrier frequency is an integral multiple of the output frequency, but this does not have to be the case if the carrier frequency is high enough – say a couple of hundred times higher than the maximum output frequency. This means the carrier frequency can be fixed, making synthesis a lot simpler.

Textbooks generally show the switching signals derived from analog circuits that compare a triangle-shaped carrier wave to a sinusoidal reference, but in reality these days, they will almost certainly be produced by software running in a microcontroller.

Pre-computed sine values are typically stored in a lookup table, and each carrier period, the appropriate one is extracted, multiplied by a scaling factor (the modulation depth) to set the amplitude, and loaded into a timer configured as a PWM generator.

In the case of three-phase inverters, you have to add some third harmonic to the reference sinusoid or the output amplitude will be limited. For a fuller explanation, see the "Variable Speed Drive for Induction Motors" article we published in the December 2025 and January 2026 issues. It provides a good overview of how this type of modulation can be implemented in a microcontroller.

Current control

In the case of current-source converters, such as in photovoltaic systems, it is preferable to control the inverter's output current rather than its voltage. There are a couple of ways to do this, both shown in Figs.6(c) & (d). Both of these assume there is a current transducer measuring i_{LOAD}.

The compensator-based current controller shown in Fig.6(c) uses a PWM modulator identical to the one discussed above, except that the reference input is not a fixed sinusoid. Instead, the PWM modulator is 'wrapped' in a current control loop that compares the load current to a sinusoidal reference and drives the modulator via a loop compensator to reduce the current error to zero.

This control method has the advantage of a fixed switching frequency, but the current ripple can vary widely.

You can also use hysteretic modulation, like in Fig.6(d), analogous to current-mode control in DC-DC converters. The switches are controlled in such a way that the load current always remains within a hysteresis band $\pm i_H$ around the reference. This control method provides fixed current ripple and built-in current limiting, but it does mean that the switching frequency is variable.

This might not be a problem for solar inverters, for example, which operate over a very narrow range of output frequencies.

Harmonic elimination

Really large (multi-megawatt) or high-voltage DC-AC converters generally don't use Mosfet or IGBT switches. Instead, they use gate-turnoff (GTO) thyristors, which come with voltage ratings up to 4.5kV and can handle currents up to 4kA or more.

Their switching frequency is limited to something less than 1kHz, so pulse-width modulation is not generally practical. Instead, a technique known as harmonic elimination is used.

The 'tri-state' waveform shown in blue in Fig.1, like all of the waveforms we have seen, is 'half-wave symmetric'. This just means that the second half-wave looks like the first reflected in the horizontal axis. This type of waveform only has odd harmonics, so the largest amplitude harmonic after the fundamental is the third. Its amplitude will be $v_{LOAD(3)} = (4V_{SRC} \div 3\pi)\cos(3\delta)$.

If we set δ to π/6, the amplitude of the third harmonic drops to zero. In fact, all multiple-of-three harmonics (n = 3, 9, 15...) become zero. Furthermore, you can add additional zero-state pairs of specific length at specific angles to eliminate other harmonics and their multiples.

With three such sets of zero-state 'notches' in the right places, you can eliminate the 3rd, 5th, 7th and 9th

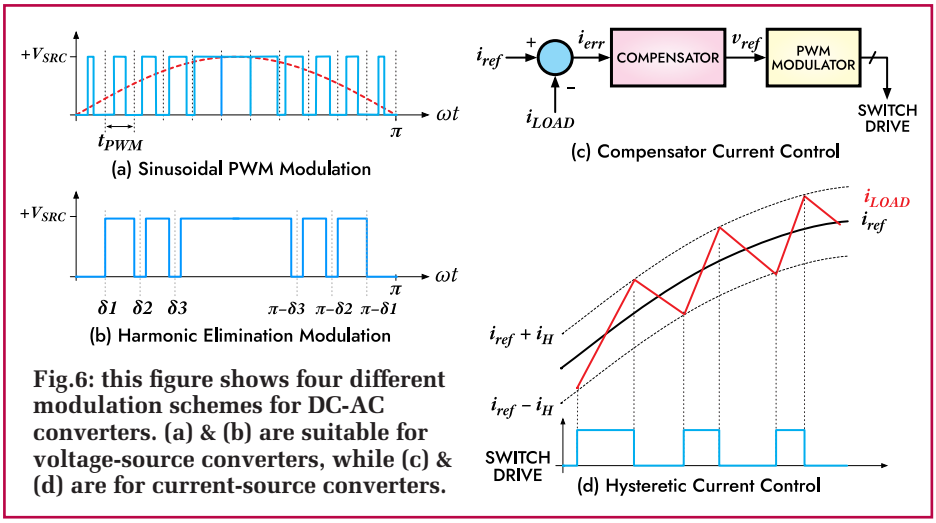


Fig.6: this figure shows four different modulation schemes for DC-AC converters. (a) & (b) are suitable for voltage-source converters, while (c) & (d) are for current-source converters.

State	Switches	v_a	v_b	v_c	v_α	v_β
x_0	2, 4, 6	−½	−½	−½	0	0
x_1	1, 4, 6	+½	−½	−½	+2/√6	0
x_2	1, 3, 6	+½	+½	−½	+1/√6	+√3/√6
x_3	2, 3, 6	−½	+½	−½	+2/√6	+√3/√6
x_4	2, 3, 5	−½	+½	+½	−2/√6	0
x_5	2, 4, 5	−½	−½	+½	−1/√6	−√3/√6
x_6	1, 4, 5	+½	−½	+½	+1/√6	−√3/√6
x_7	1, 3, 5	+½	+½	+½	0	0

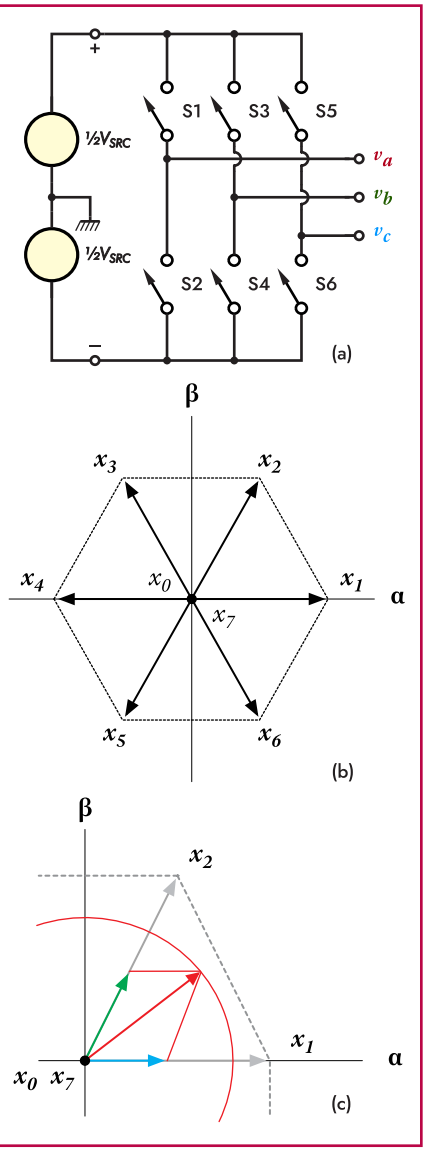


Fig.7: space vector modulation maps a set of three-phase sinusoidal waveforms to a single rotating vector in the αβ plane. You can synthesise a three-phase voltage set by determining how much time to spend in each state.

harmonics with a switching frequency just five times the fundamental. The largest harmonic present in the output will be the 11th. This is shown graphically for one half-cycle in Fig.6(b).

We already know that in a balanced three-phase system, the 3rd harmonic and its multiples are already zero, making the 5th harmonic the highest one present in the waveforms shown in Fig.5. In this case, the first harmonic elimination zero-state pair is therefore placed to eliminate the 5th harmonic and its multiples.

Three pairs of zero-state notches can eliminate all harmonics below the 13th in three-phase systems.

The main downside of harmonic elimination is that by fixing the zero-state angles, you can't use δ to control the output voltage. Converters in the multi-megawatt range are normally driven from a dedicated transformer and use tap changers to regulate voltage.

Space vector modulation

The science-fiction sounding space vector modulation (SVM) is a modulation technique used in balanced three-phase inverters, especially in motor drives. It takes advantage of the symmetry in these systems to simplify the calculations necessary to synthesise a desired set of sinusoidal waveforms.

While it may simplify the modulator, understanding it can be a bit of a mind-bender, so we are going to need some background.

It is possible to represent any set of three variables, like three-phase voltages at some instant in time, as a point in 3D space. Each of the variables defines a length along one of the three orthogonal axes. This set of three values is called a vector because it describes a point in 3D space that is some specific direction and distance from the origin.

For physical space, we usually label the coordinates x, y and z, but for our three-phase system, we will name the axes a, b and c, representing each phase. Three values (v_a, v_b, v_c) define a vector $\boldsymbol{v}_{abc}$ that represents all three-phase voltages at any instant in time. We use a bolded $\boldsymbol{v}$ to indicate that this is a vector.

In the case of balanced three-phase systems, we have a further constraint. At any moment in time, the sum of the phase voltages must be zero. This severely limits the range of values that the vector describing those variables can take. In fact, $\boldsymbol{v}_{abc}$ must lie on a plane passing through the origin and tilted such that its normal vector (a vector standing perpendicular to the plane) passes through the point (1, 1, 1).

Don't worry if this is hard to picture; we are about to simplify matters.

With quantities confined to this plane, it is natural to assign a new pair of orthogonal axes as if 'looking down' the normal vector. These axes are traditionally labelled α and β, and the plane is unsurprisingly called the alpha-beta plane. On this plane, we can describe a three-value three-phase vector as a point with coordinates (x_α, x_β).

The transformation between the two systems is known as a Clarke transform. This is named for its inventor, Edith Clarke, who among her many distinctions was the first female American professor of electrical engineering when she joined the University of Texas in 1947.

If we take a set of three-phase sinusoids and map them to the αβ plane, they correspond to a point rotating in a circle around the origin. The speed of rotation is related to the frequency, and the radius is related to the amplitude. We now have all the tools we need to explore space-vector modulation.

A three-phase inverter like that shown in Fig.7(a) can produce eight possible switch states, which produce the three-phase voltages shown in the adjacent table. The phase voltages are referenced to an imaginary 'Neutral' point at half of the source voltage and shown in the table normalised to V_{SRC}.

The last two columns in the table show how each of these switch states maps onto the αβ plane using the Clarke transformation. Don't worry too much about the weird √6 values. What is important is that if you plot these points on the αβ plane, you get the result shown in Fig.7(b).

Each non-zero state maps to a vertex of a hexagon, and the two zero states map to the origin.

Any point inside the hexagon represents a set of balanced phase voltages that we could synthesise. Fig.7(c) shows how, using an example from the triangle with vertices x_1, x_2 and x_0x_7.

To synthesise the voltage represented by the red vector, we need to be in each of these three states for an

appropriate proportion of time, so the result 'averages out' to the desired voltage set.

The length of the blue vector relative to x_1 defines the proportion of time we need to spend in state x_1, while the length of the green vector relative to x_2 defines it for the x_2 state. The remaining time is spent in one or the other of the zero-states. In the example, the fraction of x_1 is around 0.3 and the fraction in x_2 is around 0.4, leaving 0.3 of the time spent in the zero states.

If we want to synthesise a set of three-phase sinusoidal voltages, we just rotate the vector around the origin, tracing out the red circle/arc. Over each switching period (T_{SW}), the red vector advances by some angle proportional to the output frequency. Typical implementations start and end each switching period in a zero-state to minimise output harmonics.

There will therefore be three transitions per switching period: zero to state A, state A to state B and state B to zero. Each transition only requires one switch to be changed if you select the right zero-states.

The sequence of states is always the same as you rotate around the circle, so it can be programmed in advance. Only the time spent in each state has to be calculated (or looked up in a table) in real-time. This is more efficient code-wise than three-phase sinusoidal PWM, although the speed and capability of today's microprocessors makes this advantage less important than it once was.

Like every modulation technique, space vector modulation has plenty of variations and options. It's a pretty deep rabbit-hole if you want to go exploring.

Getting practical

I thought it would be interesting to dig into the design of a real DC-AC converter by looking at an aspect of converter design that is often overlooked: calculating the thermal losses in the switching elements. This is far from simple in the case of DC-AC converters with pulse-width modulation and inductive loads, as you will see.

I will use the IGBT bridge from the Variable Speed Drive Induction Motors article mentioned earlier as an example.

This three-phase converter uses DGTD65T15H2TF 650V 30A IGBTs with integral freewheeling diodes, switching at 15.625kHz. There are two kinds of losses that we have to consider: conduction losses that occur when the IGBT or diode is on, and switching losses that occur in the IGBT as it switches on and off.

Conduction losses are the product of the current through, and the voltage across, the IGBT and diode when turned on. The manufacturer provides graphs to show how these quantities are related, reproduced in Fig.8. These characteristics are not linear, and in the case of the IGBT, vary with gate drive voltage.

Conduction losses get worse with higher temperature, so I have assumed the maximum 175°C junction temperature to be safe.

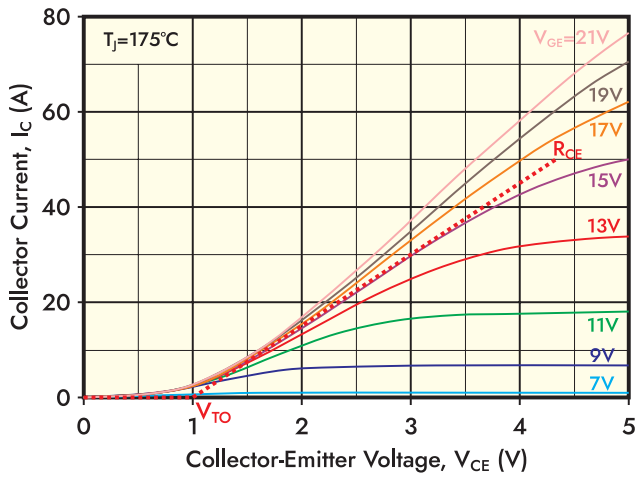
Superimposed on the charts is a dotted piecewise linear approximation that we will use to calculate conduction losses. We will assume that the IGBTs' V_{CE} is a combination of a fixed drop, V_{TO}, plus a voltage that increases linearly with current – effectively an on-resistance, R_{CE}. This converter uses a 15V gate drive, so I have set this resistance to match the appropriate curve.

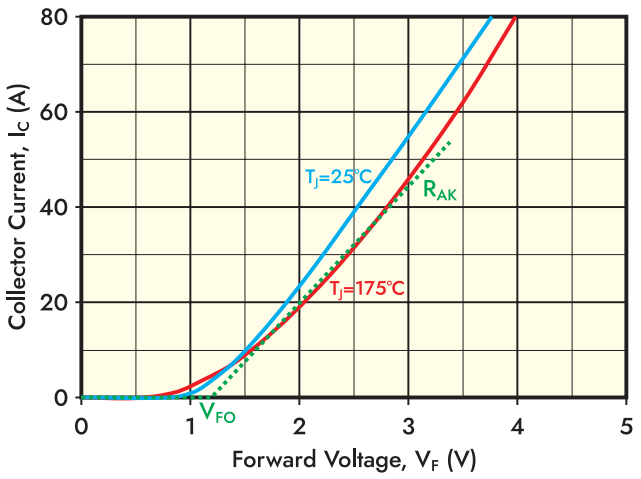
In this case, we can read off $V_{TO} \approx 1V$ and $R_{CE} \approx 90m\Omega$. We can do a similar thing with the diode characteristic, giving $V_{FO} \approx 1.2V$ and $R_{AK} \approx 40m\Omega$.

If the current through the IGBT and diode is i_c, the instantaneous conduction losses will be $P_{I(COND)} = V_{TO}\, i_c + R_{CE}\, i_c^2$ and $P_{D(COND)} = V_{FO}\, i_f + R_{AK}\, i_f^2$. The current is an AC quantity, so we have to find the average power over one cycle. This requires us to integrate these expressions over one cycle and divide by 2π.

Since the switching frequency is high and the load is inductive, the inverter current will be sinusoidal, given by the equation $i = I_{PK} \cos(\theta - \Ø)$, where Ø is the phase angle between voltage and current. This current will flow in the IGBT when the switch is on, and in the diode when the switch is off.

The proportion flowing in the IGBT is determined by the duty cycle, which is in turn defined by the phase angle and the modulation depth m, according





$$P_{I(COND)} = V_{TO}\, I_{PK}\left(\frac{1}{2\pi} + \frac{m \cdot \cos\phi}{8}\right) + R_{CE}\, I_{PK}^2\left(\frac{1}{8} + \frac{m \cdot \cos\phi}{3\pi}\right)$$

$$P_{D(COND)} = V_{FO}\, I_{PK}\left(\frac{1}{2\pi} - \frac{m \cdot \cos\phi}{8}\right) + R_A\, I_{PK}^2\left(\frac{1}{8} - \frac{m \cdot \cos\phi}{3\pi}\right)$$

Fig.8: calculating the conduction losses in an inverter bridge is not a simple undertaking. It starts with a piece-wise linear approximation of the forward characteristics and ends with the expressions below the graphs.

to the expression $D = \frac{1}{2}(1 + m\cos[\theta])$. The proportion flowing in the diode is determined by 1 – D. Plugging all this together makes for a pretty nasty integral, but I have shown the results below the graphs in Fig.8.

The total conduction loss for one IGBT/diode is the sum of these two equations. The terms involving the modulation depth m and load power factor cos(Ø) are interesting. Both can vary from zero to one, and the effect of reducing either of them is to move losses between the IGBT and the diode, while the total stays roughly constant.

This makes intuitive sense because in each case, there is less IGBT on-time and more diode freewheeling time

Switching losses are (thankfully) a little easier to calculate. Switching loss depends mainly on capacitance, so the energy expended each transition is related to the square of the voltage. The data sheet provides the switching energies for each transition, helpfully at 175°C, and at 400V, a little above the maximum we can expect, so comfortably conservative.

These energies are E_{ON} = 342µJ and E_{OFF} = 288µJ. With sinusoidal PWM, you can show the switching losses will be $P_{I(SW)} = (E_{ON} + E_{OFF}) f_{SW} \div \pi$.

Using values from the variable speed drive in single-phase mode (I_{PK} = 10A, PF = 0.95, m = 1, f_{SW} = 15.625kHz) gives conduction losses of 4.8W for the IGBT and 0.6W for the diode, and switching losses of 3.1W, for a total worst-case loss of 8.5W per device.

There are four active devices, so the total is 34W. For the three-phase case (I_{PK} = 6A, PF = 0.85, m = 1, f_{SW} = 15.625kHz), the power dissipation works out to be 5.8W per device and, coincidentally, 34W for the six active devices. That's close to 98% efficiency for the DC-AC converter part (there are additional losses in the rectifier).

You will notice that the IGBT switching losses are approaching the conduction losses in magnitude at a switching frequency of just over 15kHz. In many applications, we want switching frequencies much higher than this, so switching losses can become a huge concern in high-power converters.

The next instalment

In part seven, we will take a look at resonant or soft-switching converters, which can minimise or even eliminate switching losses. PE

DOWNLOAD PE BACK ISSUES

COMPLETE YEAR collections
From just £5.95 each

VOL 28 – January to December 1999 – £5.95
VOL 29 – January to December 2000 – £6.95
VOL 30 – January to December 2001 – £7.95
VOL 31 – January to December 2002 – £7.95
VOL 32 – January to December 2003 – £7.95
VOL 33 – January to December 2004 – £7.95
VOL 34 – January to December 2005 – £8.95
VOL 35 – January to December 2006 – £8.95
VOL 36 – January to December 2007 – £8.95
VOL 37 – January to December 2008 – £8.95
VOL 38 – January to December 2009 – £10.95
VOL 39 – January to December 2010 – £11.95
VOL 40 – January to December 2011 – £12.95
VOL 41 – January to December 2012 – £14.95
VOL 42 – January to December 2013 – £15.95
VOL 43 – January to December 2014 – £16.95
VOL 44 – January to December 2015 – £17.95
VOL 45 – January to December 2016 – £19.95
VOL 46 – January to December 2017 – £22.95
VOL 47 – January to December 2018 – £27.95
VOL 48 – January to December 2019 – £27.95
VOL 49 – January to December 2020 – £29.95
VOL 50 – January to December 2021 – £32.95
VOL 51 – January to December 2022 – £34.95
VOL 52 – January to December 2023 – £34.95
VOL 53 – January to December 2024 – £34.95

FIVE-YEAR collections – 60 issues!
From just £14.95 each

PE2832: Jan 99 to Dec 03 – £14.95
PE2933: Jan 00 to Dec 04 – £14.95
PE3034: Jan 01 to Dec 05 – £14.95
PE3135: Jan 02 to Dec 06 – £14.95
PE3236: Jan 03 to Dec 07 – £14.95
PE3337: Jan 04 to Dec 08 – £14.95
PE3438: Jan 05 to Dec 09 – £19.95
PE3539: Jan 06 to Dec 10 – £25.95
PE3640: Jan 07 to Dec 11 – £25.95
PE3741: Jan 08 to Dec 12 – £25.95
PE3842: Jan 09 to Dec 13 – £33.95
PE3943: Jan 10 to Dec 14 – £33.95
PE4044: Jan 11 to Dec 15 – £33.95
PE4145: Jan 12 to Dec 16 – £35.95
PE4246: Jan 13 to Dec 17 – £37.95
PE4347: Jan 14 to Dec 18 – £38.95
PE4448: Jan 15 to Dec 19 – £39.95
PE4549: Jan 16 to Dec 20 – £42.95
PE4650: Jan 17 to Dec 21 – £44.95
PE4751: Jan 18 to Dec 22 – £44.95
PE4852: Jan 19 to Dec 23 – £44.95

20 YEARS OF PE!
AMAZING BUNDLE OFFER
Every issue from 1999 – 2018
240 magazines for just
£69.95

PE2832: Jan 99 to Dec 03 PLUS!
PE3337: Jan 04 to Dec 08 PLUS!
PE3842: Jan 09 to Dec 13 PLUS!
PE4347: Jan 14 to Dec 18

Download from:
www.electronpublishing.com

Teach-In 2026

by Mike Tooley

World of Wireless – An Introduction to Radio and Wireless Technology, part 12.11

Long-distance radio communications

In Part 10 last month, we introduced short-range wireless devices (SRD) and showed how these versatile chips and modules enable data transmission and reception between remote devices without the need for a wired connection. Our *Hands-On* constructional project featured a simple RF field strength indicator that can be used to make relative measurements up to 450MHz.

This month, we bring our *Teach-In* to a close by providing you with some ideas that will take you further into the world of radio and wireless technology. Our final *Hands-On* project is a high-gain RF preselector.

Shortwave listening

Shortwave listening is a fascinating pastime enjoyed by many people all over the world, bringing news, documentary features, entertainment and culture to you. Thanks to the ionosphere, you can listen to scheduled broadcasts from distant countries. Shortwave listeners (SWL) tune their sets (almost any HF receiver) to signals in bands designated for international broadcasting.

Shortwave bands are often described by the wavelength in metres. They include:

- The **49m band (5.9-6.2MHz)** is usually good at night, delivering strong continental stations, as well as those from further afield, including Asia and Central America. It is less active during the day.
- The **41m band (7.2-7.5MHz)** is good during the hours of darkness. It is popular with far-eastern

Fig.11.1: shortwave receivers are widely available.

Fig.11.2: a second-hand communications receiver can be a good option for serious shortwave listening.

broadcasters, such as Radio China International on 7.415MHz. This band has a relatively narrow bandwidth of just 250kHz.

- The **31m band (9.4-10MHz)** is a firm favourite with shortwave broadcasters, suitable for use during both the day and night. Because you will nearly always be able to find interesting signals, this is a good band to get started. WWV standard time signals are broadcast on 10.000MHz.
- The **25m band (11.6-12.1MHz)** can yield good signals during the day, but is less useful at night. It's suitable for medium and long distances, particularly in the summer. Try the BBC World Service on 12.095MHz.
- The **22m band (13.6-13.9MHz)** has similar characteristics to the 19m and 25m bands. It's useful when the 19m band is closing and the 25m band has yet to open.
- The **19m band (15.1-15.8MHz)** is suitable for daytime listening, with reception possible over long distances. It's usually active in the summer months from late morning until late afternoon.
- The **16m band (17.5-17.9MHz)** is less used than other bands, and generally only viable in the summer months when the MUF is high.

As explained earlier in Part 1 (in the November 2025 issue), HF radio propagation is subject to daily and seasonal variations, with solar activity having a major impact, particularly at the upper end of the HF spectrum. Due to these variations and the need to beam signals to different continents and areas, broadcasters schedule their services with regular changes to frequency as well as language.

For example, KBS World Radio in South Korea provides the following services:

- To Japan from 01:00 to 03:00 in Japanese on 11.810MHz
- To SE Asia from 08:00 to 10:00 in English on 9.770MHz
- To Europe from 15:00 to 17:00 in English on 9.515MHz
- To Europe from 19:00 to 20:00 in French on 6.145MHz.

The changes to the schedule are designed to cater to ionospheric variations as well as the three different geographical target areas. Note that all times are quoted in UTC/GMT (not local time, although GMT coincides with UK local time during the winter).

Shortwave receivers

Shortwave receivers are widely available at costs ranging from less than £20 to more than £1000. Fig.11.1 shows some affordable options. The Roberts R862 employs conventional analog tuning with nine switched shortwave bands.

Covering the AM, FM and shortwave broadcast bands, the modestly priced Retekess V115 uses digital synthesis and has the benefit of a digital frequency display. Shortwave coverage is 4.75-21.85MHz but the radio can't be used for SSB reception (essential for communication but not required for broadcast reception).

The more expensive XHDATA D-808 caters for FM, AM, SSB (USB [upper] and LSB [lower] selectable) as well as AM air-band coverage. This type of receiver caters to multiple listening needs.

If you are lucky enough to have a bigger budget, a second-hand communications receiver like the Kenwood R-2000 shown in Fig.11.2 could be a good option. This receiver covers 150kHz to 30MHz and 118-174MHz with a plug-in VHF converter.

It incorporates several useful features such as a signal-strength meter (S-meter), a switched RF attenuator, scanning, and various clock and timer functions. Similar receivers are regularly advertised on second-hand websites at prices ranging from around £75 to £500.

Finally, if you have the right skills, it's possible to build your own shortwave receiver. Fig.11.3 shows the author's home-constructed shortwave receiver that can tune over 4-10MHz based mainly on modules described in this *Teach-In* 12 series. The SSB Shortwave Receiver project published in our

Fig.11.3: the author's home-made receiver covers the busy 49m, 41m and 31m shortwave bands.

May & June 2026 issues covers the range of 3-30MHz.

Getting started!

1. Select a radio that covers as many of the shortwave broadcast bands as possible. Your receiver should be able to tune from around 5MHz to 18MHz. If the receiver is a conventional type with analog tuning, it should preferably have several switched 'band spread' ranges. A digital frequency display will make tuning easier. Since the channel spacing for AM shortwave broadcasting is either 5kHz or 10kHz, a fine-tuning step of 5kHz (or better) is essential.
2. If the receiver has an external antenna connector, it's worth using it. A simple end-fed wire or short dipole antenna (*Teach-In* 12.5, March 2026) will usually work well, but you need to keep it away from interference sources such as switch-mode power supplies, TV receivers, monitors and computers. An external antenna fed with a length of coaxial cable will invariably prove vastly superior to a short telescopic or ferrite rod antenna.
3. Get started on the 41m and 31m bands, where you will find signals at almost any time of day or night. Having gained some listening experience, move on to explore the less predictable 25m and 19m bands.
4. Decide on a target area and use online resources to investigate current broadcasting schedules, countries and languages. Databases of shortwave broadcasting can be found at **https://shortwavedb.org**, **https://shortwave.live**, **https://shortwave.am**, as well as **https://shortwaveschedule.com** and **https://new.hfcc.org**
5. Regularly check propagation conditions, taking into account the time of day, season and current solar activity. Useful sources are **www.listenersguide.org.uk/swl/reception/**, **https://hf.dxview.org**, and **www.hamqsl.com/solar.html** Although the last two are designed primarily for radio amateurs (two-way communication), they are also useful for shortwave listeners.
6. Keep a log of stations heard, noting the date, time (UTC), signal strength, language and content. This information will help inform future activity and provide you with an invaluable record of what you've heard as well as when and where you heard it!

Fig.11.4: a typical amateur radio station with equipment supporting operation in all amateur bands from 1.8MHz to 440MHz.

Amateur radio

Amateur radio is a sociable hobby enjoyed by millions of people worldwide. Radio amateurs are licensed to use a wide range of frequency bands for non-commercial purposes, notably 'self-training' in radio and wireless technology. This normally involves one-to-one communication with other radio amateurs, many of whom can be far afield.

A typical amateur radio station is shown in Fig.11.4, with equipment for operation from 1.8MHz to 440MHz. Digital control is used for the main HF transceiver, allowing it to be operated from a PC interface (Fig.11.5).

Like shortwave broadcasting, the frequency spectrum allocated to radio amateurs is divided into several discrete bands, each with unique properties and uses. Some of the most popular bands (those that you are likely to find in everyday activity) include:

- The 80m band (3.5-3.8MHz) is popular for local and regional communication, with contacts of up to 400km by day and over 1000km at night. Conditions are generally better in the winter months, during the periods of darkness where intercontinental operation is possible. The band can sometimes be noisy due to thunderstorms and interference from local noise sources.
- The 40m band (7.0-7.2MHz) is widely used, providing signals throughout the day and night. During the day, ground-wave propagation supports dependable regional communication, while after dark, ionospheric propagation enables long-distance (DX) working. The band is also popular for international contests.
- The 20m band (14.0-14.35MHz) is popular worldwide and renowned as a 'DX band'. It regularly provides opportunities for

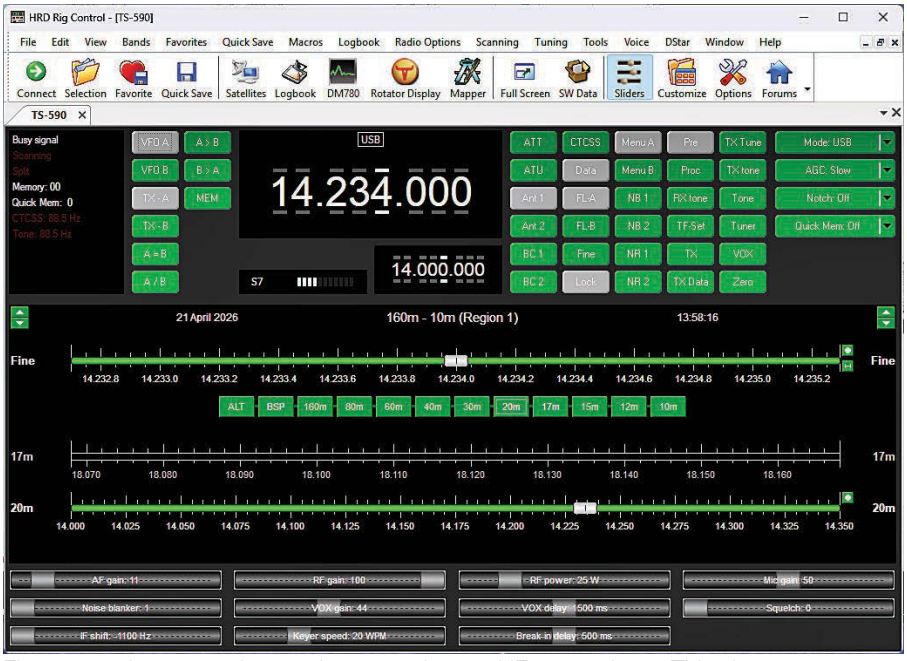

HRD Rig Control - [TS-590]
File Edit View Bands Favorites Quick Save Macros Logbook Radio Options Scanning Tuning Tools Voice DStar Window Help
Connect Selection Favorite Quick Save Satellites Logbook DM780 Rotator Display Mapper Full Screen SW Data Sliders Customize Options Forums
TS-590
Busy signal
Memory: 00
Quick Mem: 0
USB
14.234.000
14.000.000
S7
ATT CTCSS Menu A Pre TX Tune Mode: USB
ATU Data Menu B Proc TX tone AGC: Slow
Ant 1 FL-A NB 1 RX tone Tone Notch: Off
Ant 2 FL-B NB 2 TF-Set Tuner Quick Mem: Off
BC 1 Fine NR 1 TX VOX
BC 2 Lock NR 2 TX Data Zero
21 April 2026
160m - 10m (Region 1)
13:58:16
Fine
ALT BSP 160m 80m 60m 40m 30m 20m 17m 15m 12m 10m
17m
20m
AF gain: 11
RF gain: 100
RF power: 25 W
Mic gain: 50
Noise blanker: 1
VOX gain: 44
VOX delay: 500 ms
Squelch: 0
IF shift: -1100 Hz
Keyer speed: 20 WPM
Break-in delay: 500 ms

Fig.11.5: software can be used to control many HF transceivers. This shows a transceiver being set for operation at a frequency of 14.234MHz in the HF 20m band.

Fig.11.6: a typical ‘field-day’ contest where all exchanges are logged using software.

intercontinental contacts, with the ionosphere playing a major role. The band is usually open during daylight hours but usually not when it’s dark. The band also has a great deal of contest, data-mode and regular SSTV activity. Fig.11.6 shows a typical ‘field-day’ station where contest exchanges are logged using PC software.

- The 15m band (21.0-21.45MHz) can be excellent for long-distance communication, depending on solar activity. During the peak of the solar cycle, it’s possible to contact amateur stations all over the world and at good strength. At solar maximum, the band often remains open from early morning to late afternoon. Conversely, at solar minimum, openings tend to be extremely sparse, and the band can be very quiet.
- The 10m band (28-30MHz) supports occasional long-distance communication during periods of exceptional solar flux. At other times, the sporadic E-layer offers a chance for sky-wave contacts at distances of typically 1000km or more (see Part 1, November 2025).
- The 2m band (144-146MHz) is a popular VHF band used for a variety of purposes, including local FM voice chats, repeater operation, satellite communication and emergency networks. Equipment is plentiful and inexpensive (particularly low-power handheld radios). Antennas are reasonably compact and suitable for vehicle mounting. Under normal propagation conditions, ranges vary from around 20km to around 100km depending on location and antenna gain (see Part 5, March 2026). Occasionally, irregular phenomena such as tropospheric ducting or temperature inversions can extend working distances considerably. Local nets and contests are regularly held on this band, and well-located repeaters help extend the range of hand-held and mobile transceivers.
- The 70cm band (430-440MHz) is excellent for local hand-held and mobile communication due to the availability of low-cost transceivers (see Fig.11.7). In many countries, the band is well-supported with local repeaters (both digital and analog) that extend the working range. Digital repeaters are often networked, making it possible to contact stations on the other side of the world using only a hand-held radio. The coverage of a modestly located local repeater, GB7WS, is illustrated in Fig.11.8. The effect of topography is more severe at UHF than it is at VHF, with the coverage of 70cm repeaters tending to be smaller than can be achieved with comparable equipment on the 2m band.

Fig.11.7: a dual-band ‘walkie-talkie’ can be ideal for local contacts, both direct and working through repeaters.

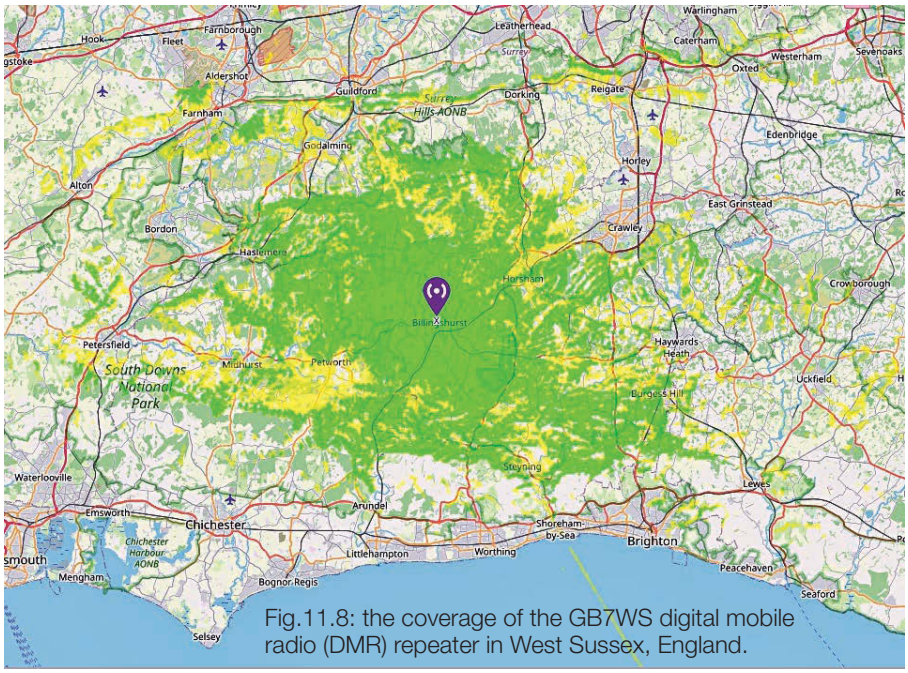


Fig.11.8: the coverage of the GB7WS digital mobile radio (DMR) repeater in West Sussex, England.

Amateur operating modes

Amateur radio operators can experiment with numerous operating modes. The choice of mode depends on several factors, including the equipment available, the prevailing operating conditions and the type of contact. Modes that are suitable for long ‘rag-chew’ conversations are unsuitable for short contacts where only basic information and signal reports are exchanged.

Popular modes like SSB, CW and FM are used regularly by most radio amateurs. Others, keen to push the boundaries and exploit leading-edge technology, make use of low-power digital modes. Popular operating modes currently used by radio amateurs include:

- CW (continuous wave) is on/off keyed Morse code, the most basic and enduring mode used by radio amateurs. It makes minimal equipment demands and outperforms other methods of communication, particularly when propagation conditions are poor. The mode remains popular for its simplicity, efficiency and ability to achieve long-distance contacts (DX) with minimal power and extremely narrow bandwidth. Most CW operators now use paddles and automatic keyers instead of the straight Morse keys that were popular more than half a century ago. Software packages are available to assist with the task of coding and decoding Morse (see Fig.11.9). That said, there's no substitute for the ears of an experienced CW operator!
- AM (amplitude modulation) is a 'legacy' mode used mainly by vintage radio enthusiasts. A double-sideband AM transmission requires twice the bandwidth of an SSB transmission, so the mode is inefficient in terms of spectrum space. Furthermore, the power present in the carrier is effectively wasted; the carrier tells you nothing except that the transmitter is switched on! This wasted power results in further inefficiency. For these two reasons, AM is now rarely used for regular communication.
- SSB (single sideband) has largely replaced AM. In SSB, one sideband is removed (either the lower or upper) and the carrier is suppressed. This increases the spectral efficiency and reduces the power demand, making SSB the preferred voice mode for HF communication as well as VHF long-distance (DX) communication. SSB reigns supreme for contest and 'field-day' voice mode operation.
- FM (frequency modulation) offers excellent quality audio and inherent noise rejection, making it ideal for local VHF and UHF communications. It is widely used in handheld and mobile transceivers as well as most repeaters. FM is ideal for local conversations and for use on the move.
- RTTY (radio teletype) uses frequency-shift keying (FSK) to transmit simple text messages. Originally requiring bulky and noisy mechanical Teletype machines, RTTY has enjoyed a resurgence in popularity with the advent of computers and digital interfaces. RTTY offers robust performance at speeds greater than CW, so the mode remains popular for contest working as well as for general HF contacts and DX-pedition stations.
- SSTV (slow-scan television) enables the transmission of images over radio. Using low data rates, pictures are sent line-by-line and decoded by software, allowing operators to exchange visual information such as drawings, cartoons, photographs and QSL cards (see Fig.11.10). The MMSSTV software shown in the figure can be freely downloaded from **https://hamsoft.ca/pages/mmsstv.php**
- PSK (phase-shift keying) is a popular mode for digital communication, supporting casual keyboard-to-keyboard chat and long-distance (DX) operation. PSK operates in a very narrow bandwidth and is thus efficient in spectrum use. The mode is resilient to noise and propagation disturbances and many contacts are made with relatively low power (often less than 30W).
- Olivia is another popular digital mode that performs well under poor conditions. Olivia uses robust error correction and configurable bandwidth settings, making it ideal for long-distance communication.
- Other 'weak signal' modes have become increasingly popular in recent years, including FT8, JT65, WSPR and VarAC (see Fig.11.11). Advanced algorithms and error correction are used to

Fig.11.9: a low-power transceiver being used to make a general (CQ) call on the 15m band. The transceiver is using software to send and receive Morse code.

Fig.11.10: slow-scan television (SSTV) is a popular way of sending images at slow rates using narrowband HF communication.

decode marginal signals, enabling contacts over extremely long distances with minimal power.

Getting started!

1. Contact your national society as well as the regulatory body in your country to obtain current information concerning amateur radio licensing. Membership in a national society is usually quite inexpensive; special rates may be available to young people. Societies provide regular publications together with news and plenty of constructional articles, often aimed at beginners. The Radio Society of Great Britain (RSGB) in the UK has an excellent book service targeted at radio amateurs at all levels.
2. Local clubs and societies invariably welcome new members, particularly those who are just starting out. Contact the secretary of your local club or society to see what support can be offered and whether training is available in your area. Most clubs have a regular program of talks, demonstrations and 'field-day' events.
3. Online training as well as online examinations are available in some countries (including the UK). It is well worth researching this at an early stage and contacting the organisers for examination and course dates.
4. Gain experience by listening to a variety of different communications between radio amateurs. Start by listening on the popular 40m and 20m bands where you will usually find plenty of activity. If you have a VHF/UHF handheld transceiver, tune into the repeaters in your area. An internet search will provide you with callsign, location information and coverage maps.
5. Make an antenna! This valuable exercise will not only provide you with a feel for antenna construction but also provide you with something useful that will help with listening and eventually also for making two-way contacts. Several of the antennas described in Part 5 (March 2026) could make useful projects. For example, a simple dipole for the 20m band or a vertical quarter-wave ground plane for the 2m band.

LoRa

Last month, we introduced short-range wireless devices (SRD) that can be used to support communication between sensors, actuators and microcontrollers at distances of typically up to 20m.

SRD provides us with a neat and flexible alternative to conventional wiring and cabling. Devices can be quickly and easily moved and relocated without the need for extensive rewiring. LoRa takes this one step further, catering for longer distances with the ability to add a layer of network connectivity.

LoRa stands for **lo**ng-**ra**nge communication and it is based on chirp spread spectrum (CSS) technology, where signals are transmitted in the form of frequency-modulated bursts that sweep across a range of frequencies.

CSS enables robust data transmission in difficult and noisy environments, with significant advantages compared with the WiFi technology that we also covered last month.

By spreading the transmission over a wider bandwidth, signals are less susceptible to interference and other disturbances. LoRa performs well in a built environment where wireless signals can be severely attenuated and suffer scattering and undesirable multipath propagation.

In addition, the relatively low data rates and minimal duty cycle make LoRa ideal for remote control and monitoring systems. Depending on national legislation and regulations, LoRa operates in various 'sub-GHz' bands, including 433MHz, 868MHz and 915MHz (see Table 11.1). It is also available in the 2.4GHz ISM band, where higher data rates are possible at the cost of working range.

LoRaWAN

To enable network operation, media access protocol (MAC) protocol can be applied to LoRa so that multiple devices can be interconnected to form part of a wide area network (WAN).

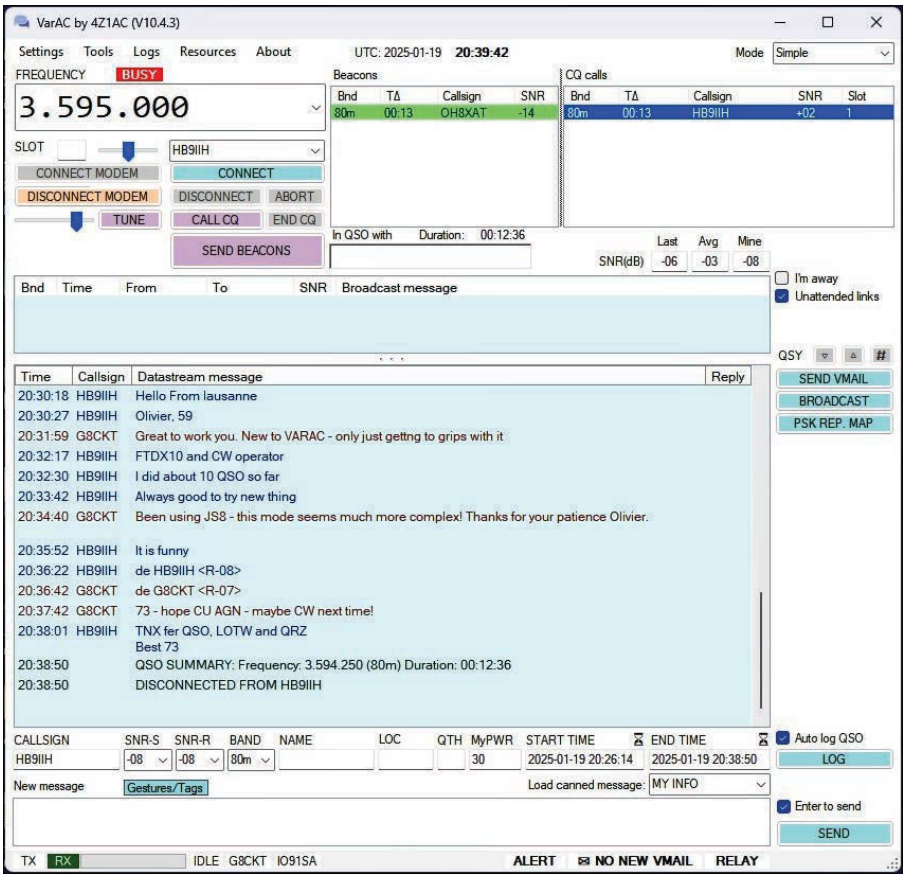

VarAC by 4Z1AC (V10.4.3)
Settings Tools Logs Resources About
UTC: 2025-01-19 20:39:42
Mode Simple
FREQUENCY BUSY
3.595.000
Beacons
CQ calls
SLOT
HB9IIH
CONNECT MODEM
CONNECT
DISCONNECT MODEM
DISCONNECT ABORT
TUNE
CALL CQ END CQ
SEND BEACONS
In QSO with Duration: 00:12:36
SNR(dB)
I'm away
Unattended links
Bnd Time From To SNR Broadcast message
QSY
SEND VMAIL
BROADCAST
PSK REP. MAP
Time Callsign Datastream message Reply
20:30:18 HB9IIH Hello From lausanne
20:30:27 HB9IIH Olivier, 59
20:31:59 G8CKT Great to work you. New to VARAC - only just gettng to grips with it
20:32:17 HB9IIH FTDX10 and CW operator
20:32:30 HB9IIH I did about 10 QSO so far
20:33:42 HB9IIH Always good to try new thing
20:34:40 G8CKT Been using JS8 - this mode seems much more complex! Thanks for your patience Olivier.
20:35:52 HB9IIH It is funny
20:36:22 HB9IIH de HB9IIH <R-08>
20:36:42 G8CKT de G8CKT <R-07>
20:37:42 G8CKT 73 - hope CU AGN - maybe CW next time!
20:38:01 HB9IIH TNX fer QSO, LOTW and QRZ
Best 73
20:38:50 QSO SUMMARY: Frequency: 3.594.250 (80m) Duration: 00:12:36
20:38:50 DISCONNECTED FROM HB9IIH
CALLSIGN SNR-S SNR-R BAND NAME LOC QTH MyPWR START TIME END TIME
HB9IIH -08 -08 80m 30 2025-01-19 20:26:14 2025-01-19 20:38:50
Auto log QSO
LOG
New message
Gestures/Tags
Load canned message: MY INFO
Enter to send
SEND
TX RX IDLE G8CKT IO91SA
ALERT NO NEW VMAIL RELAY

Fig.11.11: a conversation between the author and a Swiss radio amateur using the powerful VarAC digital mode on the 80m amateur band. The software has reported an average signal-to-noise ratio of -3dB (ie, there's more noise than signal)!

Table 11.1: frequencies and standards for sub-GHz LoRa

Region	Freq. range	Standards/ regulations
Australia	915-928MHz	AU915-928
China	470-510MHz, 779-787MHz	CN470-510, CN779-787
Europe & UK	863-870MHz, 433MHz	EU863-870, EU433, CEPT Rec. 70-03
India	865-867MHz	IN865-867
Japan	920-923MHz	ARIB STD-T108
New Zealand	915-928MHz	SRD - GURL
USA	902-928MHz	US902-928

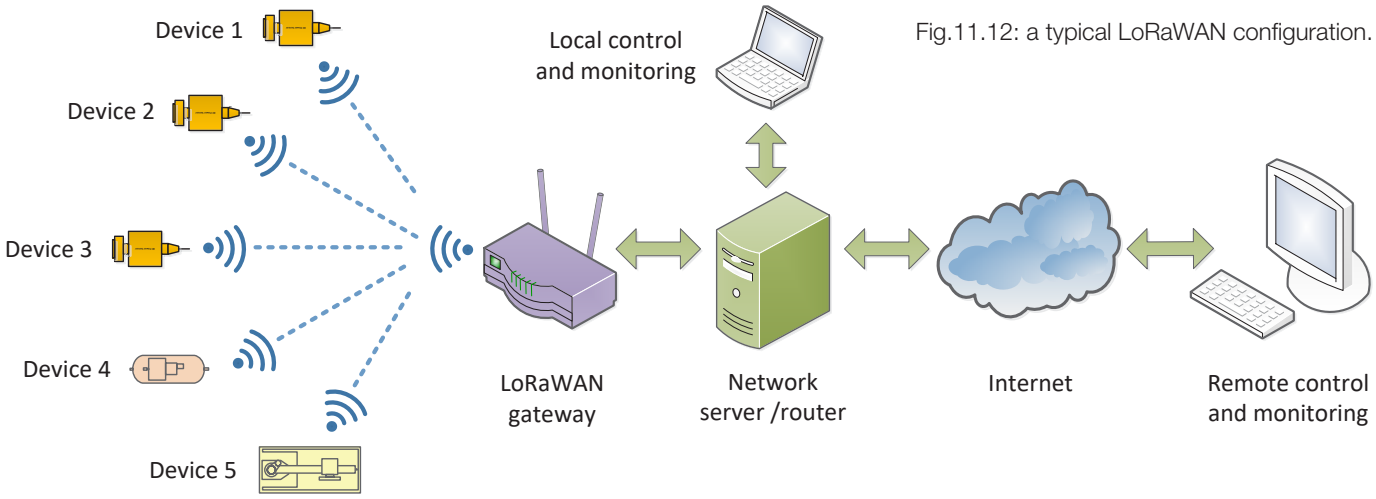


Fig.11.12: a typical LoRaWAN configuration.

Developed and maintained by the LoRa Alliance, the LoRaWAN protocol supports communication between devices and servers that can be several kilometres apart in an urban area and up to 10km in a less-obstructed rural environment.

Typical LoRaWAN applications include environmental monitoring, smoke and haze detection, proximity sensors, wild animal conservation, smart farming, monitoring food production, water purification, object and vehicle tracking (without the need for GPS) and even the support of patients suffering with dementia. The list is virtually endless!

Fig.11.12 shows a typical LoRa configuration in which multiple devices communicate via a LoRaWAN gateway. The arrangement supports local control and monitoring via a network server as well as an internet connection via a router, enabling devices to be controlled and monitored from almost anywhere in the world.

Getting started!

1. Consider whether 433MHz or 868MHz is appropriate for your project. While signals at 433MHz usually travel further and penetrate well inside a built environment, 868MHz can often perform better in a real-world scenario due to the higher permissible power. In addition, 868MHz is widely used across Europe and is well supported. Both bands are licence-exempt, but if you also happen to be a radio amateur, 433MHz is probably the better choice.
2. Consider future needs as well as current requirements. It's always best to allow for expansion; your system should be designed so that new functions and devices can be easily added. You will find plenty of information on the web, but a good starting place is **https://pemag.au/link/accc**
3. Be careful when purchasing sensors and other devices. It's essential to ensure that they are compatible with the LoRa standard that you're using! For example, a device designed for use in the USA (under US902) will not be usable in Europe (under EU868). Since many dubious 'grey market' devices have appeared recently, it's not only important to check their origin, but also their conformance with the required standards.
4. Observe power and duty cycle limits. Under EU868, the standard output power is 25mW (14dBm) ERP with a duty cycle that's usually set to 1%. By minimising the power and duty cycle, there's less chance of interference, so multiple IoT systems can co-exist in the same spectrum space.
5. Purchase one of the popular LoRa development modules. These are widely available and usually well-documented. Many are based on the ESP32 microcontroller, but others are available for the Raspberry Pi, including the low-cost Pi Pico (Fig.11.13). Your development board should comprise a LoRa transceiver together with the necessary microcontroller, interface, antenna connector and voltage regulator.
6. A good antenna is often key to obtaining maximum range, but siting is also important and is all too often overlooked.

Aircraft radio

Aircraft make extensive use of radio for both communication and navigation at every stage of flight, as shown in Fig.11.14. Voice transmissions to and from aircraft can be regularly heard by tuning across the VHF aircraft band (particularly if you happen to be close to an airport). Eavesdropping on them can provide you with a fascinating insight into the world of aviation.

VHF 'air-band' receivers covering

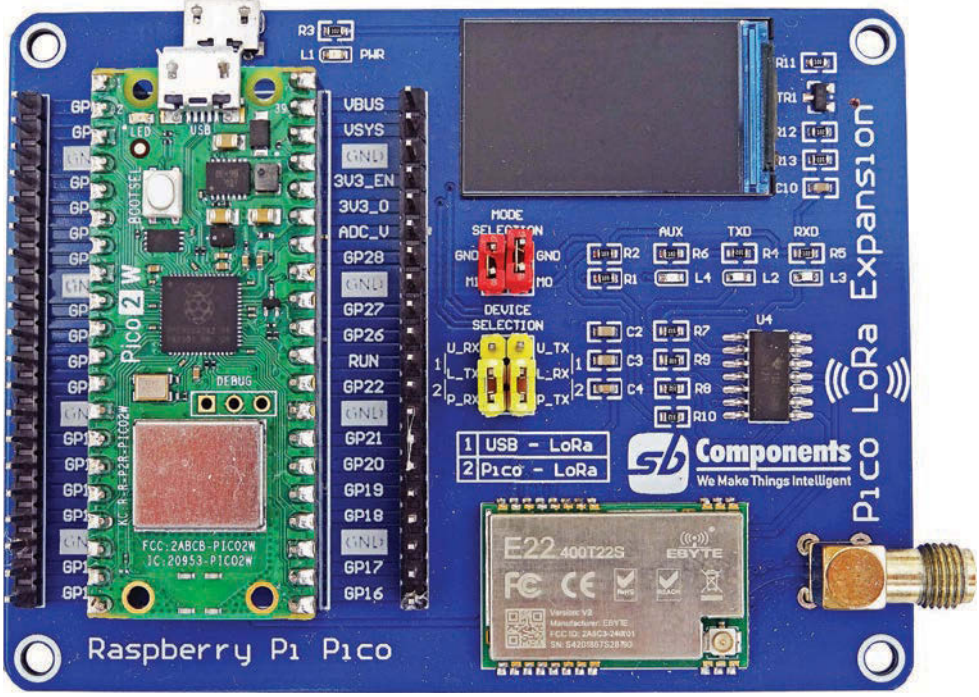
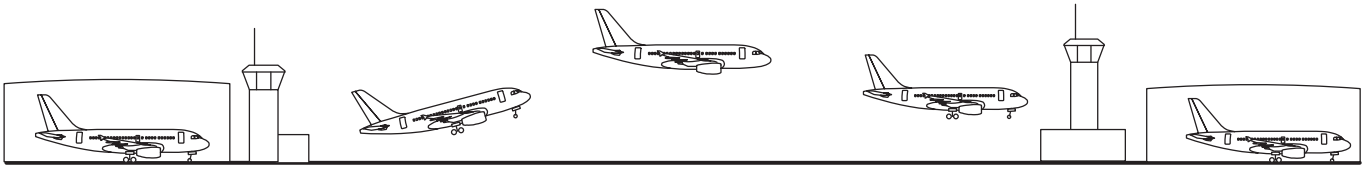


Fig.11.13: a LoRa development board based on an inexpensive Raspberry Pi Pico.



On the ground

Voice communication: VHF
Data communication: VDL

From the aircraft
Fuel data
Crew information
Link test
To the aircraft
Weight and balance data
Airport information
Flight plan
Meteorological data
Ground handling

Take-off and departure

Voice communication: VHF
Data communication: VDL

From the aircraft
Engine data.
To the aircraft
Flight plan update
Weather reports
Traffic updates

En route

Within line of sight
Voice communication: VHF
Data communication: VDL
Outside line of sight
Voice communication: HF
Data communication: HFDL and SATCOM

From the aircraft
Position reports
ETA/delay information
Weather reports
Engine information
Maintenance reports
To the aircraft
Flight plan updates
Weather reports
Oceanic clearances

Arrival and landing

Voice communication: VHF
Data communication: VDL

From the aircraft
Gate requests
Provision requests
ETA
Engine information
Maintenance reports
To the aircraft
Gate assignment
Passengers and crew data

On the ground

Voice communication: VHF
Data communication: VDL

From the aircraft
Fuel information
Crew information
Fault data
To the aircraft
Taxi information
Ground handling

Fig.11.14: aircraft use radio for communication and navigation at every stage of a flight.

the frequency range of 118-137MHz are readily available at relatively low cost. Immediately below this range (and above the range used for VHF FM broadcasting) is a band used by navigational beacons and instrument landing system systems.

The band above 118MHz is mainly used for AM voice communication, divided into channels with 25kHz and (more recently) 8.33kHz spacing. In a few countries, it's illegal to listen to aircraft communications, so it's important to check local legislation before attempting to receive VHF transmissions from either an aircraft or a ground station.

Fig.11.15 shows an aircraft in voice contact with the Final Director at London Heathrow Airport displayed using an SDR receiver (see Part 4, February 2026). Notice the signals present on other nearby channels.

Aircraft VHF radio equipment supports data communications as well as voice communications. The system used for the aircraft data link is known as Aircraft Communications Addressing and Reporting System (ACARS).

Most large civil aircraft are equipped with three VHF radios, two of which are used for voice communications, and one is used for the ACARS data link (referred to 'airline operational control communication' or AOC).

A data link terminal on board the aircraft generates downlink messages and processes uplink messages received via the VHF data link (VDL). The downlink and uplink ACARS messages are encoded as plain ASCII text.

Data is transferred from the VHF radio to the ACARS management unit at 2.4kbps using frequency shift keying (FSK). The FSK audio signal consists of two tones, one at 1.2kHz and one at 2.4kHz, depending on whether the polarity of the information bit being transmitted is the same as that of the previous bit or is different.

The phase of the tones varies linearly; there is no phase change on the transition between the two tones. This type of modulation (in which the frequency spacing between the two audio tones is exactly half the data rate) is highly efficient in terms of bandwidth and is referred to as minimum shift keying (MSK).

The MSK signal is used to modulate the amplitude of the VHF carrier (in much the same way as the voice signal). The resulting transmitted signal is double sideband (DSB) AM modulated at 2.4kbps.

High Frequency (HF) radio provides aircraft with an effective means of communication over long-distance oceanic and transpolar routes. Global voice and data communication is made possible by strategically located networked HF data link (HFDL) ground stations, augmenting VHF and SATCOM links.

Unlike the VHF air band, HF radio operates on several spot frequencies, and the channels used are non-contiguous, spread widely over the HF spectrum and interspersed with

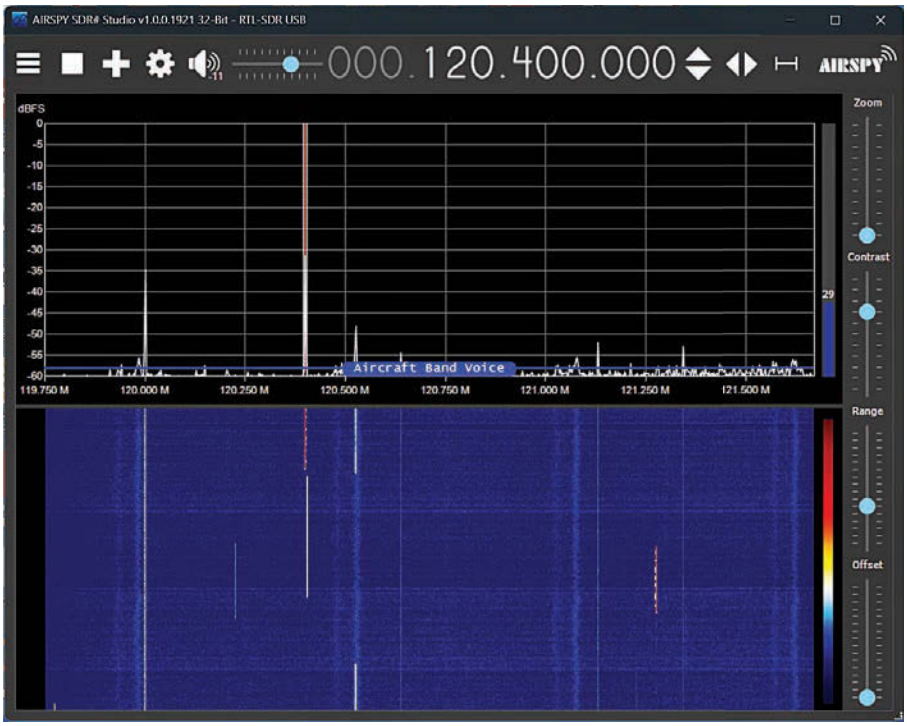


Fig.11.15: an SDR receiver being used to monitor a voice contact between an aircraft approaching Heathrow Airport at 120.4MHz. Other signals are visible too.

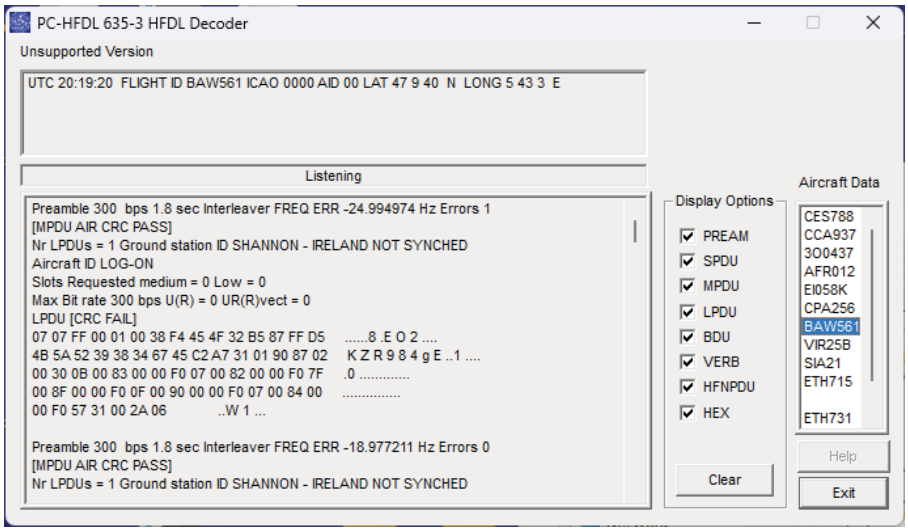

PC-HFDL 635-3 HFDL Decoder
Unsupported Version
UTC 20:19:20 FLIGHT ID BAW561 ICAO 0000 AID 00 LAT 47 9 40 N LONG 5 43 3 E
Listening
Preamble 300 bps 1.8 sec Interleaver FREQ ERR -24.994974 Hz Errors 1
[MPDU AIR CRC PASS]
Nr LPDUs = 1 Ground station ID SHANNON - IRELAND NOT SYNCHED
Aircraft ID LOG-ON
Slots Requested medium = 0 Low = 0
Max Bit rate 300 bps U(R) = 0 UR(R)vect = 0
LPDU [CRC FAIL]
07 07 FF 00 01 00 38 F4 45 4F 32 B5 87 FF D5 .....8 .E O 2 ....
4B 5A 52 39 38 34 67 45 C2 A7 31 01 90 87 02 K Z R 9 8 4 g E ..1 ....
00 30 0B 00 83 00 00 F0 07 00 82 00 00 F0 7F .0 .....
00 8F 00 00 F0 0F 00 90 00 00 F0 07 00 84 00 .....
00 F0 57 31 00 2A 06 ..W 1 ...
Preamble 300 bps 1.8 sec Interleaver FREQ ERR -18.977211 Hz Errors 0
[MPDU AIR CRC PASS]
Nr LPDUs = 1 Ground station ID SHANNON - IRELAND NOT SYNCHED
Display Options
PREAM
SPDU
MPDU
LPDU
BDU
VERB
HFNPDU
HEX
Clear
Aircraft Data
CES788
CCA937
3O0437
AFR012
EI058K
CPA256
BAW561
VIR25B
SIA21
ETH715
ETH731
Help
Exit

Fig.11.16: an HFDL transmission received at Shannon from an Airbus A320.

other services such as shortwave broadcasting, fixed point-to-point, marine, land-mobile, government and amateur services.

High Frequency Data Link (HFDL) coverage provides a highly cost-effective data link capability for aircraft following remote trans-oceanic and polar routes (where SATCOM coverage can be extremely limited) and most traffic is related to AOC.

HFDL uses phase-shift keying (PSK) at data rates of 300bps, 600bps, 1200bps or 1800bps. The data rate used depends on the signal quality and prevailing propagation conditions.

HFDL is based on frequency division multiplexing (FDM) for access to ground station frequencies and time division multiplexing (TDM) within individual communication channels.

HFDL is relatively easy to receive using modest equipment and PSK decoding software. To cope with varying propagation conditions, ground stations operate on different frequencies at different times of day.

For example, the Shannon ground station in Ireland operates on frequencies of 5.547MHz, 6.532MHz, 8.942MHz or 11.384MHz depending on the time of day.

Fig.11.16 shows an HFDL transmission received at Shannon from an Airbus A320 operating British Airways flight BA561. The transmission was decoded using the popular PC-HFDL software, and at the time of sending, the aircraft was over Eastern France.

Automatic dependent surveillance-broadcast (ADS-B)

ADS-B is a technology designed for air traffic management. ADS-B supplements existing ground-based radar systems, providing surveillance over a much wider area than might otherwise be possible. It allows aircraft to be identified in real-time with their position, speed, altitude and heading displayed without needing a conventional radar system.

By linking ground stations (eg, by satellite) it is possible for aircraft to 'see' each other as well as for their flight to be accurately monitored and tracked from the ground.

Fortunately, ADS-B technology can be very easily accessed by an enthusiast without the need for expensive professional-grade equipment. All you need is an SDR dongle, an antenna, a smartphone, tablet or PC, and some software. Fig.11.17 shows the author's phone receiving ADS-B using a low-cost SDR dongle and dual-band 7dBi antenna. The tracks of individual aircraft can be easily seen.

Fig.11.17: receiving ADS-B using a phone, a low-cost SDR ADS-B dongle and a dual-band +7dBi antenna.

Suitably equipped ADS-B enthusiasts can participate in the global Airspy Server Network, which enables multiple clients to access ADS-B data over the internet. For further information on the Airspy Server Network, see **https://airspy.com/quickstart/**

Fig.11.18 shows a map of Airspy ground stations currently active in the UK and parts of Northern Europe.

Getting started!

1. A low-cost SDR dongle (see Part 4, February 2026) will get you started with VHF aircraft listening and will allow you to receive voice signals, particularly if

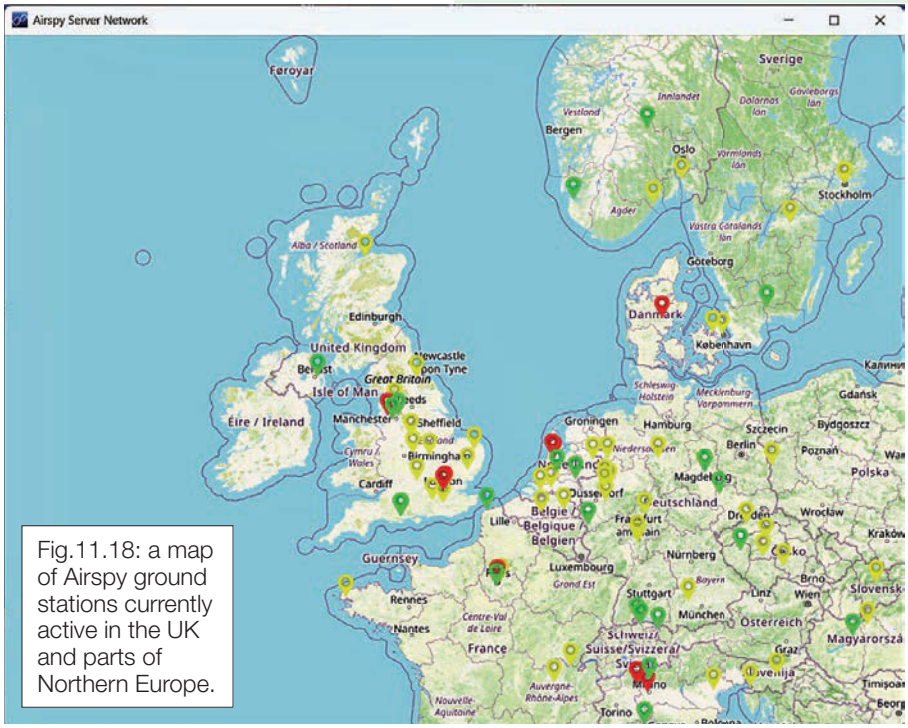


Fig.11.18: a map of Airspy ground stations currently active in the UK and parts of Northern Europe.

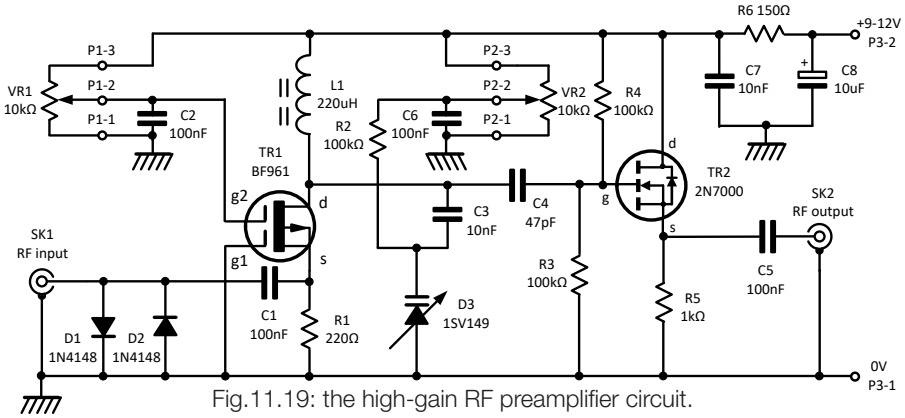
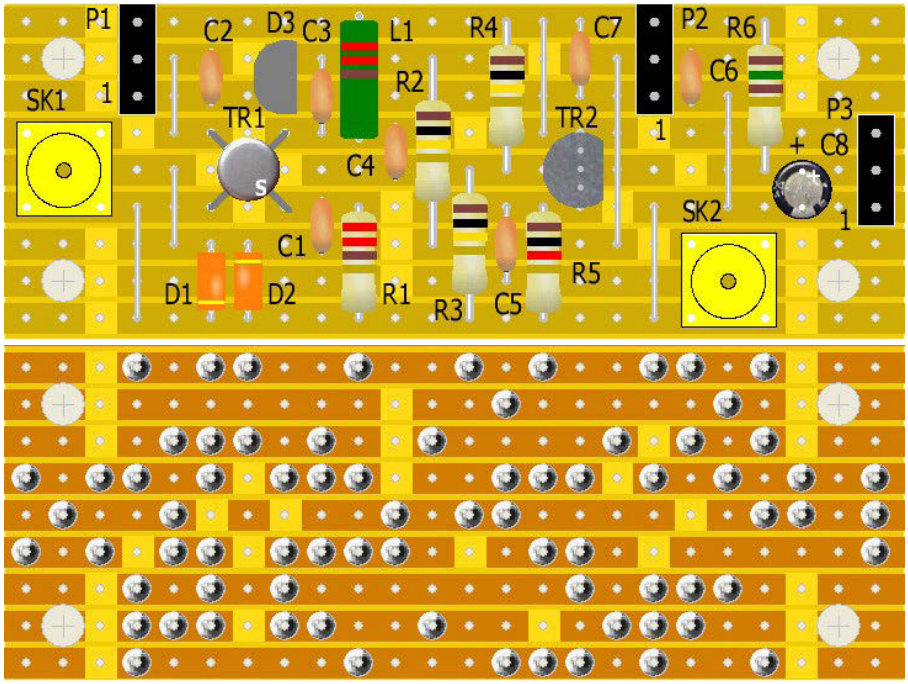


Fig.11.19: the high-gain RF preamplifier circuit.



Figs.11.20 & 21: the high-gain RF preamplifier component layout (top & bottom views).

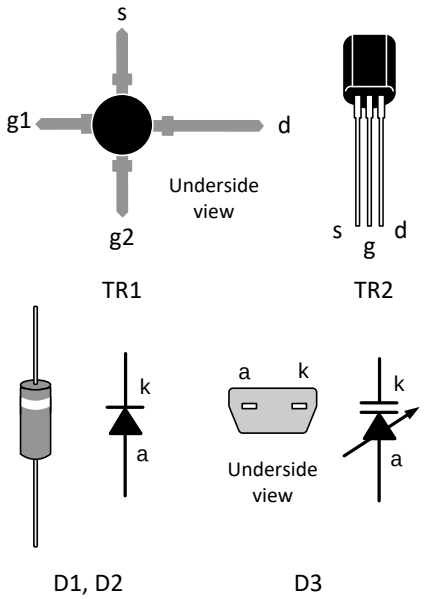


Fig.11.22: the semiconductor pin connections for the high-gain RF preamplifier.

you live anywhere near a busy airport.

2. With a simple quarter-wave vertical antenna or our *Hands-On* dipole reference antenna (Part 5, March 2026), you should have little difficulty locating signals. It's also worth using the internet to check the frequencies being used at your local airport. You should find that several are in regular use.
3. If you plan to try ADS-B reception, it's worth spending a little more to obtain an SDR designed specifically for use with ADS-B. A lower noise figure and front-end filtering can improve ADS-B performance, but results will be disappointing if you intend to use the same dongle for VHF voice and data. If you wish to receive VHF and ADS-B, you may do better purchasing a good quality general-purpose SDR.
4. ADS-B operates at 1.090GHz, so your receiving antenna should be designed specifically for this frequency. Unfortunately, our *Hands-On* dipole reference antenna (described in Part 5 of this series, in the March 2026 issue) is not suitable for operation at such a high frequency.

Hands-On: A high-gain RF preamp

This month's *Hands-On* project is a high-gain RF preamplifier that's ideal for long-distance (DX) listening on the medium-wave (MW) band. If a different range is required, just a single inductor needs to be changed.

The circuit of the RF preamplifier is shown in Fig.11.19; it comprises a dual-gate Mosfet common-gate amplifier (TR1) followed by a standard Mosfet (TR2) connected as a unity-gain source-follower.

The preamplifier is designed with a low-impedance input suitable for use with a random-length long-wire antenna, with the two anti-parallel silicon switching diodes (D1 & D2) providing over-voltage protection.

A parallel resonant circuit formed by inductor L1 and variable-capacitance diode D3 (in series with DC isolating capacitor C3) forms the load for TR1. The very high impedance of this arrangement, together with negligible loading from TR2, ensures a very high degree of selectivity.

Source-follower TR2 is suitable for driving a low-impedance output and/or a relatively long output cable run.

Sensitivity is varied by means of a front panel control, VR1, while a second front panel control, VR2, acts as a tuning control.

The preamplifier requires a supply current of less than 10mA from an external 9-12V DC source, such as a plugpack.

As with most of our *Hands-On* projects, it is built on a small piece of perforated copper stripboard measuring 25 × 64 mm, arranged as nine strips each of 24 rows. The component layout (viewed from above) is shown in Fig.11.20, while the corresponding track layout (viewed from below) is in Fig.11.21.

The required 27 track breaks can be made using a spot-face cutter or a small drill bit, while the seven links on the upper side of the board can be made using short lengths of tinned copper wire.

Take care to observe the correct polarities of TR1, TR2 and D2 (see Fig.11.22).

The RF preamplifier can be mounted in a small ABS enclosure. Suitable cases are available from various online suppliers at reasonable prices.

The enclosure used for the prototype measured 100 × 68 × 50mm. The finished stripboard is mounted on the underside of the enclosure lid, together with VR1, VR2, S1, and

the power connector, SK3, as shown in Fig.11.23.

If your receiver has an effective automatic gain control (AGC) or is already fitted with an RF gain control, there is no need to fit the sensitivity control, in which case P1-2 and P1-3 can simply be linked.

The BNC-type RF input and output connectors (SK1 and SK2) are mounted at opposite ends of the enclosure, as per Fig.11.24. To avoid the need to solder the coaxial connector, we've once again made use of panel-mounting BNC sockets with 15cm pigtail cables and SMA connectors.

With VR1 set to minimum gain, the finished preamplifier should deliver a voltage gain of 30dB at 750kHz, rising to around 35dB at 1.9MHz. Fig.11.25 shows the frequency response of the unit when tuned to 1.5MHz at the upper end of the MW band (a voltage gain of 50 is equivalent to a gain of 34dB).

From Fig.11.25, the measured bandwidth is approximately 80kHz.

The 9-12V DC supply is attached using a standard 5.5mm DC barrel connector, SK3. The front panel layout of the RF preamplifier is shown in Fig.11.26.

Table 11.2 shows the different ranges that can be obtained by changing the value of L1, while Table 11.3 provides some typical test voltages.

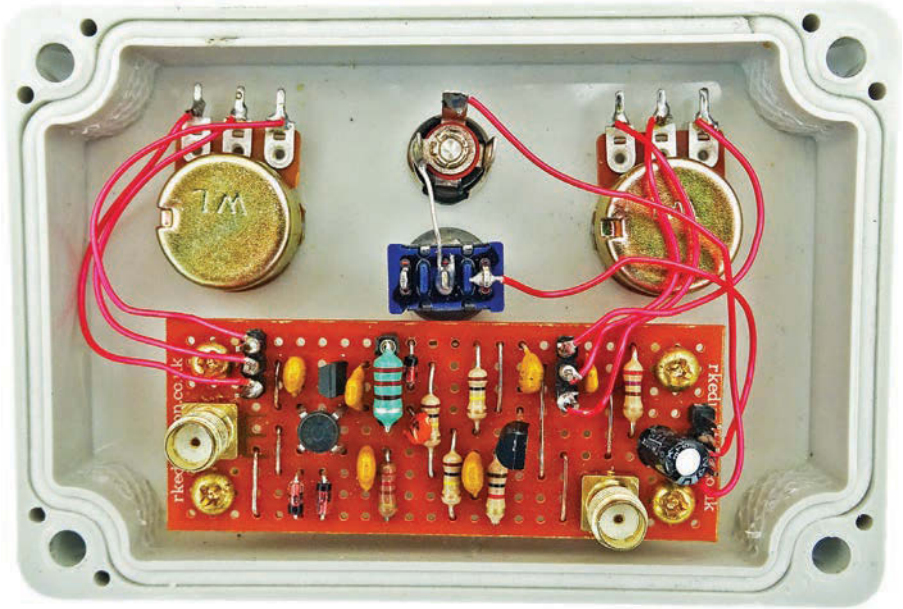


Fig.11.23: the internal wiring of the high-gain RF preamplifier.

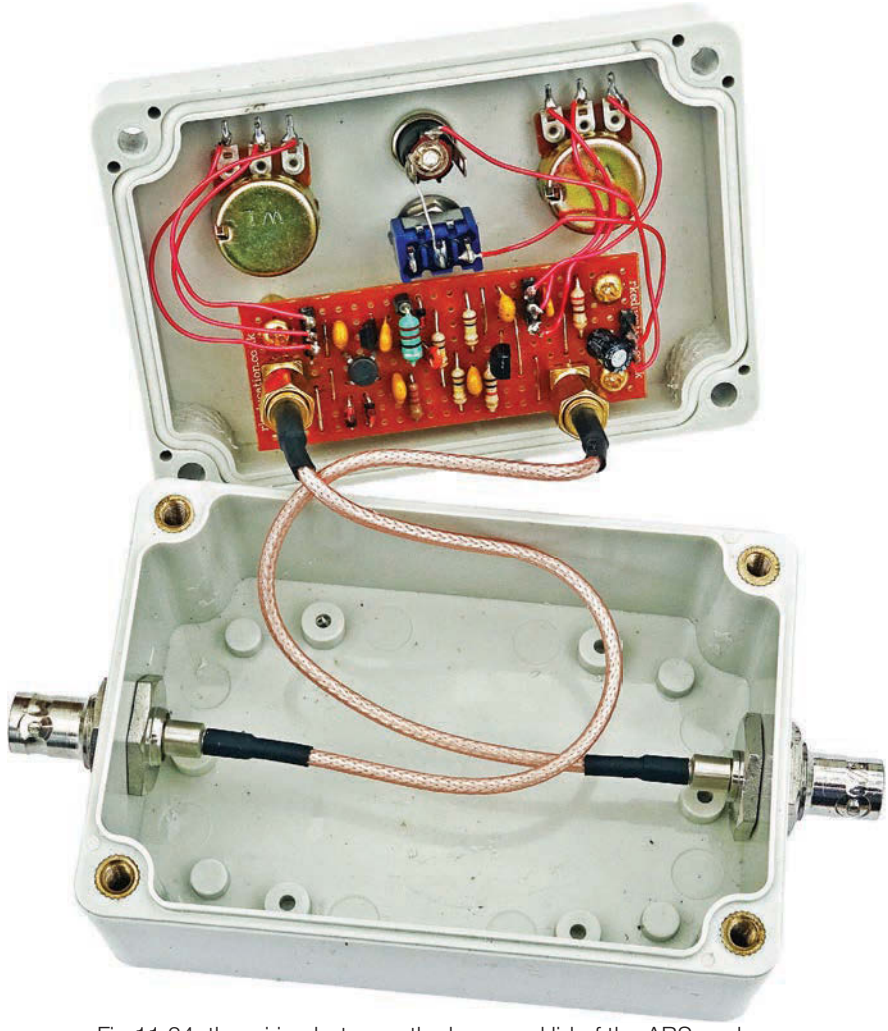


Fig.11.24: the wiring between the base and lid of the ABS enclosure.

Conclusion

With that, we come to the end of this 11-part educational series on radio and radio communications.

Over nearly a year, we have covered topics including:

- Magnetic and electric fields
- Radio signal propagation
- Atmospheric effects on radio

Table 11.2: typical frequency ranges depending on the value of inductor L1

L1 value	Minimum	Maximum
220µH	470kHz	1.9MHz
100µH	690kHz	2.7MHz
33µH	1.2MHz	4.6MHz

Table 11.3: typical test voltages for 12V DC and 9V DC supplies

Part	Pin	@9V	@12V
TR1	d	8.1V	8.4V
	g2	5.0V*	5.1V*
	g1	0V	0V
	s	0.5V	0.5V
TR2	d	8.1V	8.4V
	g	4.0V	4.2V
	s	2.3V	2.4V

* varies with the setting of VR1

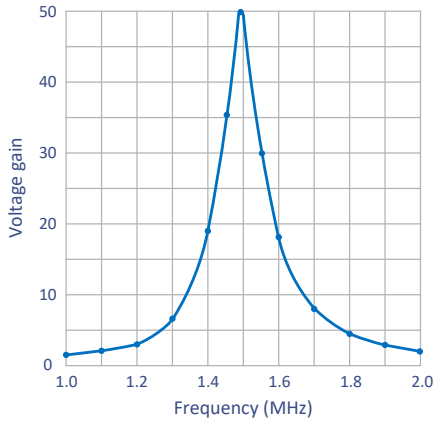


Fig.11.25: the frequency response of the high-gain RF preamplifier when tuned to 1.5MHz.

- The ionosphere
- Building a modern ‘crystal set’
- Tuned circuits
- Quartz crystals
- RF inductors and transformers
- RF semiconductors
- Modulation schemes
- Frequency conversion
- Software-defined radio
- Direct digital synthesis
- Antennas
- Baluns
- RF testing & measurement
- Spectrum analysis
- Vector network analysis
- Smith charts
- Frequency conversion
- Multipliers & mixers
- RF sources & oscillators
- Short-range RF devices
- Shortwave & amateur radio

We hope that you have found this series entertaining as well as enlightening! PE

Parts List – High-gain RF preamplifier

2 10kΩ linear variable potentiometers (VR1 & VR2)
1 220µH axial lead inductor (L1) (see Table 11.3)
2 SMA female PCB mounting connectors (SK1 & SK2)
2 panel-mounting female BNC sockets with male SMA 15cm pigtail connectors (for bringing SK1 & SK2 to the outside of the case)
1 ABS plastic enclosure (minimum 100 × 68 × 50mm)
2 small plastic pointer knobs
4 brass or nylon M3 hexagonal 5mm stand-off spacers
4 short M3 panhead machined screws
4 short M3 countersunk head machined screws
1 5.5mm outer diameter male panel-mounting 9/12V DC barrel connector
1 miniature SPST toggle switch (S1)
3 3-way 0.1-inch/2.54mm pitch pin headers (P1, P2 & P3)

Semiconductors
1 BF961 dual-gate Mosfet (TR1)
1 2N7000 Mosfet (TR2)
2 1N4148 silicon switching diodes (D1 & D2)
1 1SV149 variable-capacitance diode (D3)

Capacitors
1 10µF 16V radial electrolytic (C8)
4 100nF 50V ceramic (C1, C2, C5 & C6)
2 10nF 50V ceramic (C3 & C7)
1 47pF 50V ceramic (C4)

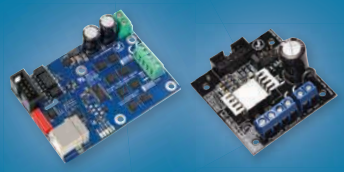
Resistors (all fixed ¼W axial, ±5% or better)
3 100kΩ (R2, R3 & R4)
1 1kΩ (R5)
1 220Ω (R1)
1 150Ω (R6)

Fig.11.26: the front panel layout of the high-gain RF preamplifier.

PoLabs
For more information or software download please visit
www.poscope.com/epe

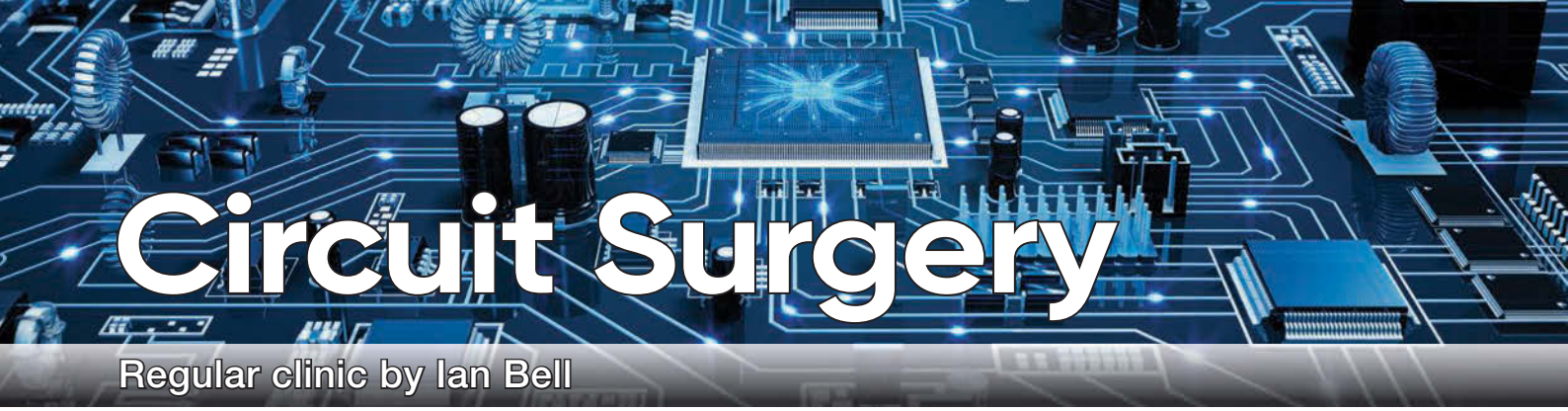
PoKeys
Connect, Control

- USB
- Ethernet
- Web server
- Modbus
- CNC (Mach3/4)
- IO
- PWM
- Encoders
- LCD
- Analog inputs
- Compact PLC

Stepper motor drivers
Drive

- up to 256 microsteps
- 50 V / 6 A
- USB configuration
- Isolated
- up to 32 microsteps
- 30 V / 2.5 A

PoScope Mega1+
PoScope Mega50
Measure

- up to 50MS/s
- resolution up to 12bit
- Lowest power consumption
- Smallest and lightest
- 7 in 1: Oscilloscope, FFT, X/Y, Recorder, Logic Analyzer, Protocol decoder, Signal generator



Circuit Surgery

Regular clinic by Ian Bell

Plant music, and using a 555 timer to measure resistance

Recently, friends who had moved away years ago returned for a visit. A few of us were chatting over old times and we recalled that they had had a device in their home that generated music when connected to a houseplant. It reacted to changes in the plant's biology, such as photosynthesis starting or ending, or the soil moisture level changing.

I was curious about how it was implemented but never got around to investigating. Having been reminded about it, I discovered several commercial 'plant music' devices and open-source designs on GitHub.

The open-source designs, and other commercial implementations, use a 555 timer IC connected to a microcontroller to measure fluctuations in the electrical resistance of the plants, or anything conductive to which it is connected.

These measurements are translated into either MIDI (Musical Instrument Digital Interface) commands or directly into synthesised music audio, delivered via a built-in speaker or a phone app. This process is often called (bio) sonification.

Using the timing behaviour of resistor-capacitor or resistor-inductor circuits is a well-established approach to measuring both component (R, L and C) values and voltage (via the dual-slope analog-to-digital converter [ADC]).

For example, time-based measurement was used in the recent *PE* Digital Capacitance Meter (December 2025). Therefore, using timing to make resistance measurements was not surprising. However, the 555 had somewhat slipped from my consciousness, despite it being one of the chips I experimented with a lot when I first became interested in electronics.

The 555 has been described as one of the most popular ICs of all time and has a place in the IEEE (Institute of Electrical and Electronics Engineers) Chip Hall of Fame for the most influential microchips in history (**https://spectrum.ieee.org/special-reports/chip-hall-of-fame/**).

After its launch in the early 1970s, it quickly became widely used in commercial applications as well as becoming very popular with hobbyists. Many billions of units have been sold.

In this article, we will start with some background on the 555 and its operation. We will then discuss its use for the resistance measurement role in the plant music device. This provides a low-cost means of acquiring resistance data for a microcontroller, but it is not the only approach available. We will look at potential-divider-based resistance measurement as a possible alternative.

Although we will discuss these techniques in the context of the plant music system, both methods are more widely applicable. These two approaches to resistance measurements are not the only possibilities, but other techniques, including those typically used in high-accuracy laboratory test equipment, will not be covered in this article.

An anniversary

The 555 was designed by Hans Camenzind (1934-2012) under contract with Signetics, a USA-based semiconductor company that was bought by Philips (now NXP) in 1975. The 555 design was completed and released in 1971; by 1972, it was being manufactured by twelve semiconductor companies. This year is the 555's 55th anniversary.

That is a good excuse to take a look at it here, with the resistance measurement application providing some focus for a chip with many uses. The IEEE Hall of Fame refers to it as "the Swiss Army knife of countless circuits".

The 555 was very popular with hobbyists and featured in many projects in the pages of magazines such as *Practical Electronics* in the first couple of decades after its release. The number of projects using 555s has reduced in more recent years because its mainstream LED flashing, timing and noise-making applications are now easily implemented using low-cost microcontrollers.

In *PE* over the last year or so, it has featured once – in the Styloclone musical instrument (June 2025), which is a retro-style project.

Despite this decline, the 555 remains a useful device for beginners to experiment with. It is straightforward to create simple, fun projects that do not need any code to be written or the use of any apps to program devices.

Furthermore, as the 555 contains a couple of comparators, a flip-flop and some output circuitry, it potentially has a wide range of uses, including as a peripheral to microcontrollers, as in plant-music circuits.

The 555 is still used in large numbers if the stock levels on distributors' websites are any indication of sales volumes. At the time of writing, a quick check of one major distributor and a few variants of the device showed stock levels of over one hundred thousand.

It is certainly still in commercial products; not long ago, I found a 555 in a bathroom extractor fan I was replacing (it provided the automatic switch-off timing).

If you are interested in the variety of projects inspired by the 555, the entrants to *Hackaday's* 555 Timer Contest (**https://hackaday.io/submissions/555-timer-contest/list**) are worth a browse. This includes one of the plant music devices mentioned earlier ("Biodata Sonification v05"). The competition was to celebrate the 555's 50th anniversary in 2021.

The 555 timer IC

The 555 can produce accurately timed one-off pulses (referred to as monostable operation) initiated by an input pulse. Pulse times can range from microseconds to minutes, with values outside this range achievable, possibly at the expense of accuracy and depending on the specific variant used.

The 555 can also be configured as a square-wave oscillator (astable

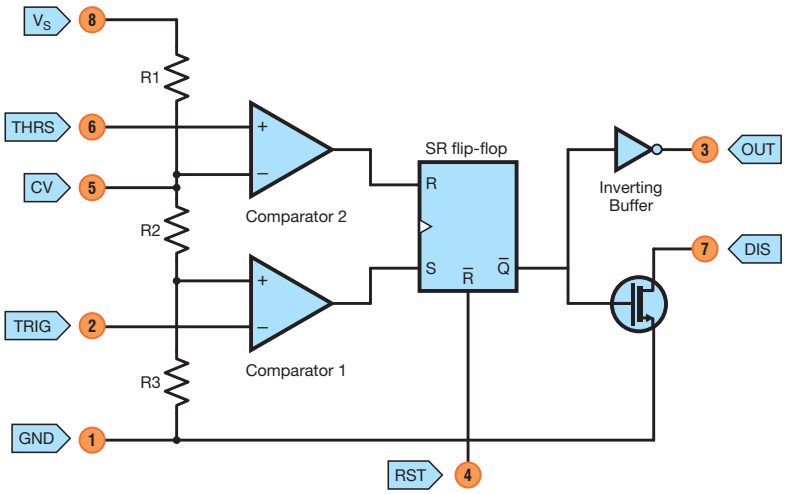


Fig.1: the simplified internal schematic of the 555 IC, with DIP pin numbers.

operation). Frequencies can range from sub-hertz to a few hundred kilohertz for the bipolar versions, and into the low megahertz for the CMOS version. The timing of both monostable and astable operation is set by external resistor-capacitor combinations and can also be modulated via a control voltage input.

There are several variants of the 555 timer:

- The NE555 is the original, which uses bipolar transistor technology.
- The LM555 is a later bipolar version that is more stable against variations in temperature and supply voltage, but it costs more.
- The LMC555 and 7555 are CMOS versions, with much lower consumption, lower output current capability and higher frequency capability, at a similar cost to the LM555.

They are all available in 8-pin DIL packages (DIP-8) as well as a number of SMD formats, such as SOIC (eg, SOIC-8).

The aforementioned devices are probably the most commonly referred-to part numbers, but there are a variety of others with different initial letters/codes relating to different manufacturers (eg, TLC555 and ICM7555). There is also the 14-pin 556, which is a dual version with two independent 555 timers in a single package (the only shared pins are for the power supply).

Bipolar versions can typically operate from 4.5-16V, with the CMOS versions running from 1.5-15V. As with most ICs, supply bypass capacitors are required close to the device to ensure correct operation, typically a 100nF ceramic capacitor.

An additional electrolytic capacitor, such as 1µF or more, may be useful where the supply leads are long, the supply is noisy, or the 555 is driving heavier loads. A small capacitor on the control-voltage pin may also be recommended to reduce reference-voltage noise. As always, consult the datasheet for the specific device you are using for exact supply ranges and decoupling requirements.

Internal details

A simplified internal circuit of the 555 timer is shown in Fig.1. The three equal-value resistors (R_1, R_2, R_3) form a potential divider, providing reference levels of ⅓ and ⅔ of the supply voltage (V_S) for the two comparators. These reference levels can be modified by applying a voltage to the control voltage (CV pin) input.

The comparator outputs are connected to a set-reset (SR) flip-flop, which in turn drives the chip's two outputs (the OUT and DIS pins). Both comparator inputs are connected to an external resistor-capacitor circuit. The time taken for the capacitor to charge or discharge past the comparator thresholds determines the length and/or frequency of the output pulses.

Comparator 1's external input is connected to the 555's trigger input (TRIG pin). If this is taken below $\frac{1}{3}V_S$, the comparator output will go high, setting the flip-flop and starting an output pulse. This is used to trigger monostable pulse generation. It also starts a new oscillation cycle in astable mode when the external timing capacitor discharges below the trigger threshold (more on that later).

Comparator 2 is connected to the threshold input (THRS pin). If this is taken above $\frac{2}{3}V_S$, the comparator output will go high, resetting the flip-flop and terminating an output pulse. The length of the pulse in monostable mode and the length of the positive part of the astable oscillation cycle is determined by the time taken for the external timing capacitor to charge above this threshold.

The main output (OUT pin) is driven by a buffer capable of sourcing tens of milliamperes (200mA for the bipolar version of the chip). The buffer is inverting and driven from the flip-flop's $\overline{Q}$ (inverted) output, so the voltage on OUT can be interpreted as a logic level equal to the state of the flip-flop.

Outputs from the bipolar and CMOS versions of the device are compatible with corresponding logic levels (TTL and CMOS respectively) when operated on suitable supplies.

The DIS (discharge) pin is primarily intended for discharging the timing capacitor. It is an open-drain N-channel Mosfet output connection on the CMOS version (as shown in Fig.1) and an open-collector NPN in the bipolar version. When the flip-flop holds a 0 (so $\overline{Q}$ is 1), the transistor is switched on and conducts to create the discharge path.

The DIS pin can also be used as second logic output by connecting a suitable pull-up resistor from DIS to the supply, or can be used where an output that only pulls down is desired (eg, allowing it to be connected in parallel with other such outputs).

The reset input (RST pin) is an active-low input for the flip-flop which overrides its other inputs, forcing the flop-flop to hold a 0, hence causing a low (0V) output on the OUT pin and switching on the transistor connected to the DIS pin.

Activating the reset input will terminate a monostable pulse or stop astable oscillation. If the reset function is not needed, the RST pin should be connected to the positive supply to prevent false resets.

Applying a voltage to the control input (CV pin) will directly set the threshold voltage for comparator 2, with the trigger voltage threshold for comparator 1 being equal to half this applied voltage due to the potential divider formed by R_2 and R_3. If the CV pin is not used, decoupling it to ground with a 10nF capacitor can often improve operation, while increasing the component count.

The MIC1555 and MIC1557 from Microchip are non-standard versions of the 555 with only five pins in an SMD SOT-23-5 package. The MIC1555 does not have the reset, discharge and control voltage pins of the original 555 but can implement both astable and monostable circuits.

The MIC1557 has the trigger and threshold pins connected together internally. It can only implement astable circuits, but has a chip-select pin to provide a low-power shutdown function. It can operate at up to 5MHz.

Monostable configuration

As mentioned, there are many circuits that can be built using the 555, but two

that all users of the device should be familiar with are the standard monostable and astable configurations. We will describe these with reference to LTspice simulations using the idealised NE555 model that is provided with LTspice. This is found in the Misc section of the component library.

Fig.2 shows an LTspice schematic of the standard 555 monostable circuit. This circuit produces a pulse of length $1.1R_AC_1$ in seconds, which with the values shown (9.1kΩ and 50nF) is $1.1 \times 9.1\times10^{3}\text{k}\Omega \times 50\times10^{-9}\text{F} = 500\mu\text{s}$.

The schematic includes a decoupling capacitor (**C3**), as required in a real circuit, but it will not have a significant effect in the simulation due to the idealised supply (**V1** source).

A low pulse on the trigger input, which must be shorter than the required output pulse length, will produce a high pulse on the output, starting at the high-to-low edge of the input signal. If the trigger input is held low longer than $1.1R_AC_1$, the output will stay low for the duration of the low level on the trigger input (the flip-flop is being constantly reset – see Fig.1).

The reset (**RST** pin) is not used in the circuit in Fig.2 so it is tied to the supply (**VS**).

Fig.3 shows the results of simulating the circuit in Fig.2. The top pane shows a single 100µs trigger pulse, while the middle pane shows the output of the 555, which is the expected 500µs pulse, starting at the low-going edge of the trigger. The lower pane shows the voltage on the **C1** timing capacitor (**ThresDis** signal, red trace).

Once the output goes high, the DIS pin is no longer shorted to ground via the transistor connected to the DIS pin. This allows **C1** to charge towards the supply voltage via **RA**, following a normal RC charging curve until the voltage reaches the comparator 2 (see Fig.1) threshold at ⅔ of the supply voltage (3.33V in this case).

If the **CV** pin is not connected to an external voltage, it will be at this $\frac{2}{3}V_S$ voltage. This is confirmed by plotting it in the lower pane of Fig.3 (cyan trace), which shows that the output switches when the capacitor voltage reaches this level.

The **CV** pin is decoupled to ground with a 10nF capacitor, as recommended, but like the supply decoupling, it is unlikely to have a significant influence on the idealised model used here.

Standard astable configuration

Fig.4 shows an LTspice schematic of the standard 555 astable circuit. This circuit produces oscillations with a period of $T = 0.693(R_A + 2\ R_B)C_1$. The

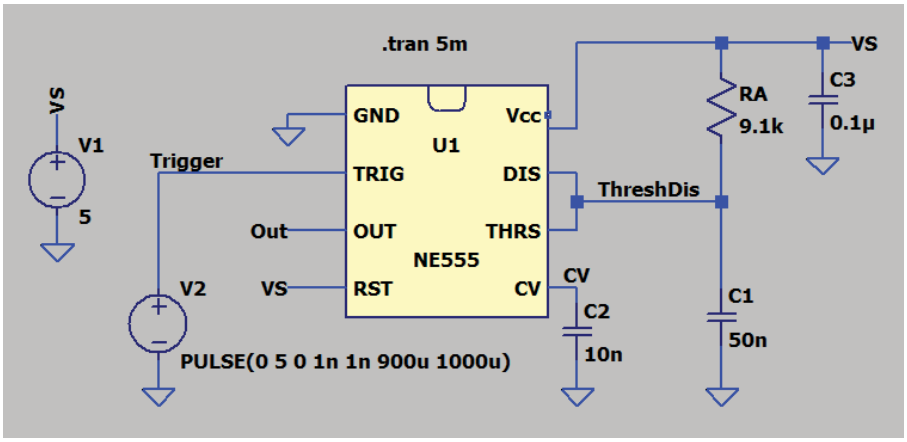


Fig.2: an LTspice schematic of a standard 555-based monostable circuit.

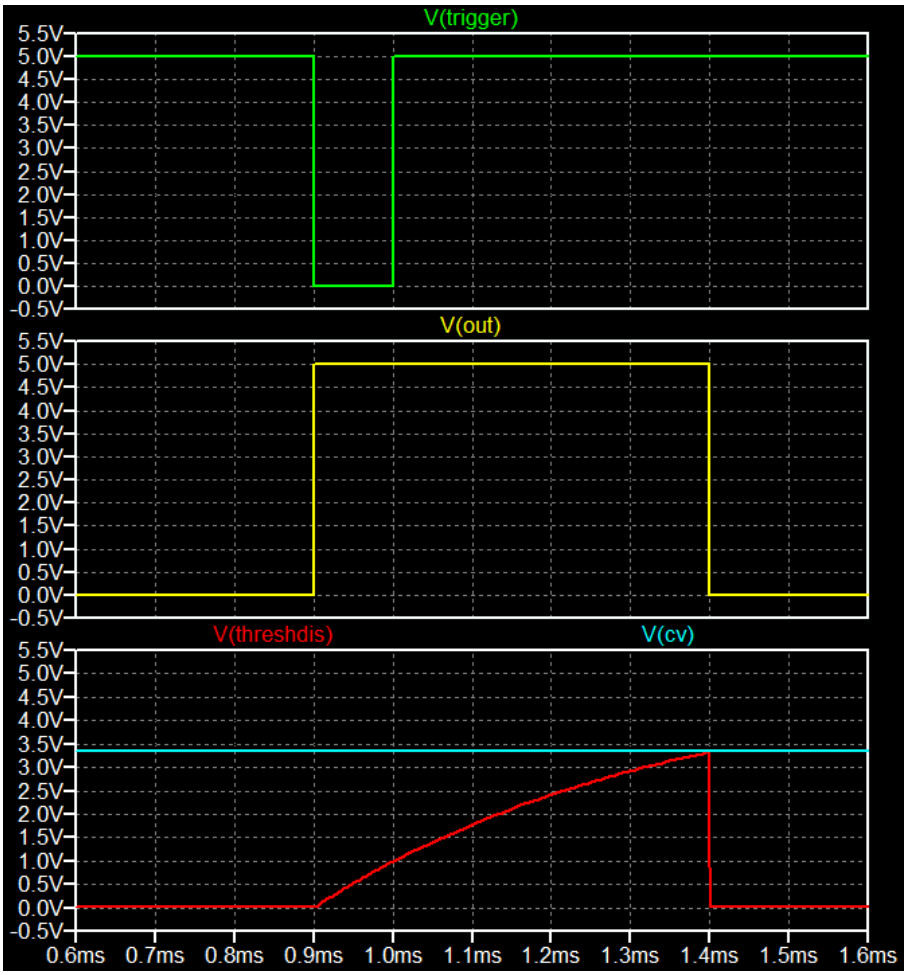


Fig.3: the simulation results from the circuit in Fig.2.

frequency (f=1/T) is:

$$f = \frac{1.44}{(R_A + 2R_B)C_1}$$

With the values shown (R_A = 7.3kΩ, R_B = 3.56kΩ and C_1 = 50nF), the frequency is f =1.44 ÷ ((7.3×10³Ω + 2×3.56×10³Ω) × 50×10⁻⁹F) ≈ 2kHz.

Unlike the monostable circuit in Fig.4, it has no external control inputs and simply oscillates continuously. As with the astable example, the reset (**RST** pin) is not used, so it is tied to the supply (V_S). The comparator inputs (**TRIG** and **THRS** pins) are tied together, with both controlling the oscillation timing. As with the monostable configuration, the supply rail and **CV** pin are decoupled.

Fig.5 shows the results of the simulation from Fig.4. The top pane shows the output of the 555, where we see the expected 2kHz square-wave oscillation. The lower pane shows the voltage on the C_1 timing capacitor (**TrigThres** signal, red trace) along with comparator 2 reference voltages – the **CV** pin voltage (at $\frac{2}{3}V_S$) (cyan trace).

For comparator 1, the threshold is $\frac{1}{3}V_S$, which is plotted directly using waveform maths (**V(vs)/3**).

With reference to Fig.1 and Fig.4, the astable operates as follows. When

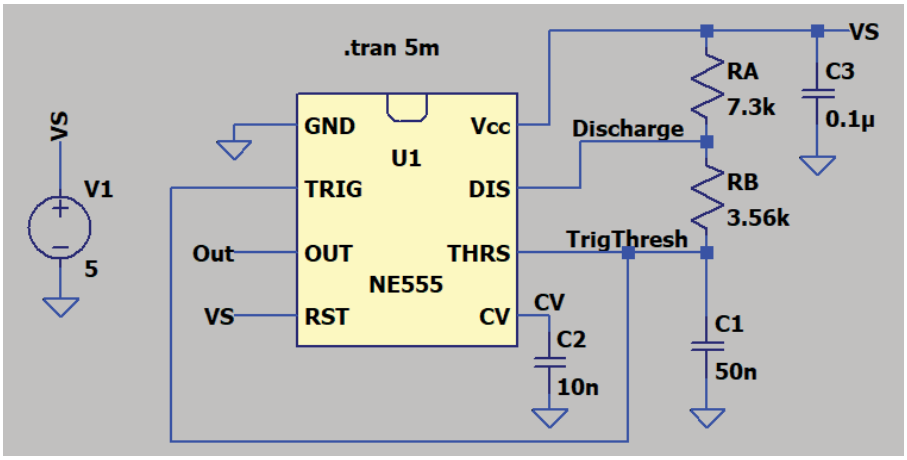


Fig.4: an LTspice schematic of a standard 555-based astable circuit.

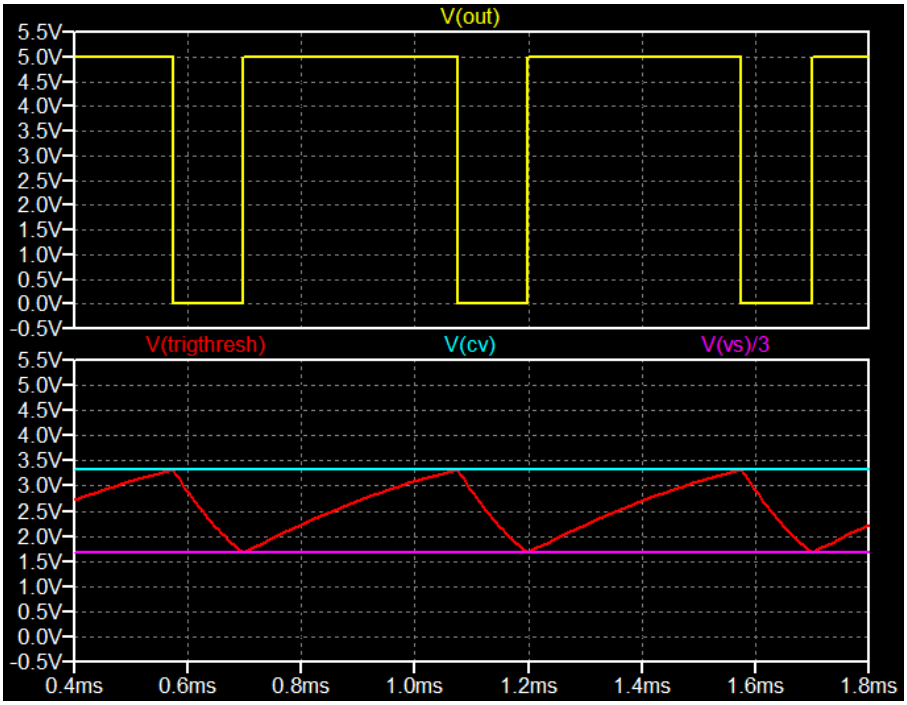


Fig.5: the simulation results from the circuit in Fig.4 with a 5V supply.

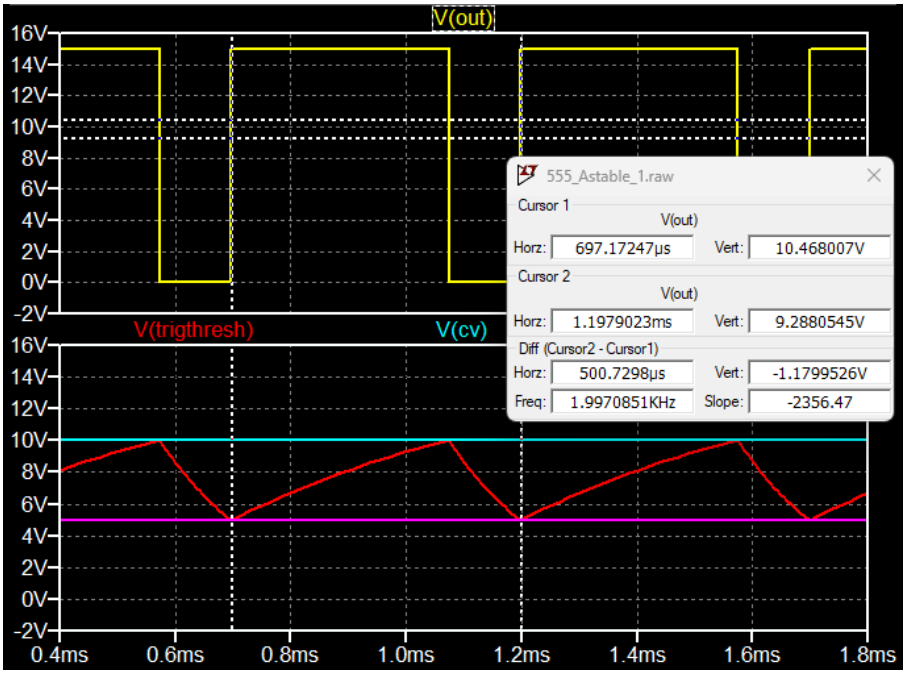


Fig.6: the simulation results from the circuit in Fig.4 with a 15V supply.

the flip-flop's Q output is high, the output (OUT pin) is high (at V_S) and the transistor connected to the discharge (DIS) pin is off, allowing the timing capacitor (C_1) to charge towards the supply voltage via **RA** and **RB**.

The voltage on the capacitor increases, following a normal RC charging curve with a time constant related to the values (R_A+R_B) and C_1. This can be seen on Fig.5; for example, from 0.7ms to 1.1ms (red trace).

When the voltage reaches ⅔V_S (cyan trace on Fig.5), comparator 2 switches on, resetting the flip-flop and bringing the 555's output low (to 0V). Now that the flip-flop holds a zero, the transistor driving the DIS pin is conducting, which will cause C_1 to discharge via R_B and the transistor. The discharge time constant is set by just R_B and C_1 as R_A is not on the discharge current path.

When the voltage on C_1 falls to ⅓V_S (magenta trace on Fig.5), comparator 2 switches on, setting the flip-flop, switching off the transistor and sending the 555's output high. This can be seen in Fig.5, for example, from 1.1ms to 1.2ms (red trace). The capacitor starts charging again, and the process described above repeats, creating the oscillation.

The charge and discharge times depend on fixed fractions of the supply voltage, so the frequency does not depend on the supply voltage. This is illustrated in Fig.6, which shows the simulation from Fig.4 rerun with the supply at 15V rather than 5V. The timing is the same. The figure also shows cursors measuring the frequency as 2kHz.

Plant music

Musicians, particularly those interested in electronic and experimental music, have been using electrical measurements of plants (and fungi) as inputs to synthesisers and other generative/algorithmic music systems since the 1970s. For example, John Lifton's "Green Music" in 1976.

A more recent pioneer is the sonic artist Mileece (**www.mileece.is/bio**), who has been developing plant music hardware and software for over 20 years. Early plant music uses systems developed by the artists themselves that are not widely available to others.

Things changed in the mid-2010s with the release of commercially available devices. The Damanhur religious community in Italy (branded as Plant Music, **www.plantmusic.com**) had been working on plant measurement and music devices since the 1970s and produced their first commercial device, the U1, in 2014.

However, it was the MIDI Sprout, designed by Sam Cusumano, whose Kickstarter launched at the same time as the U1's release, that popularised plant music due to its viral marketing and press attention.

Development of the MIDI Sprout started in 2012 with the creation of installations at the Philadelphia Museum of Art by Joe Patitucci from the Data Garden record label (**www.datagarden.org**). It was released as a product and open-source project (**https://github.com/electricityforprogress/MIDIsprout**) in 2016.

Plant Music and Data Garden have produced more up-to-date devices: the Bamboo (2018) and PlantWave (2020). The enhancements mainly concern built-in music-playing capabilities, wireless connectivity and associated phone apps, rather than the plant resistance measurement process.

There are also now several other plant music devices on the market, which are similar to these two. The Instruō Scíon (**www.instruomodular.com**) is different in that it is an analog synthesiser module which generates control voltage and gate signals; however, its design is based on the MIDI Sprout.

Plant music device operation

Fig.7 shows a simplified schematic of a plant music device, similar to the MIDI Sprout. Although the MIDI Sprout is discontinued as a product, it is the simplest version of such devices with open-source design information, so will be used as an example here.

The device is connected to the plant via a couple of flat-jawed crocodile clip or self-adhesive, sticky-pad electrodes (as used in medical applications). The plant takes the place of resistor R_B in a standard 555 astable circuit. The output from the 555 goes to a digital input pin of microcontroller, configured to trigger an interrupt when an edge (eg, a rising logic transition) occurs on the pin.

The Midi Sprout used the common ATmega328P microcontroller.

The interrupt handler calculates the time in microseconds since the previous interrupt to obtain the period (T) of the 555's oscillation. This is straightforward to implement, as microcontroller software libraries typically provide microsecond counter functions, so it is just a matter of finding the difference between the values returned by the current and previous calls to the function.

As per above, $T = 0.693(R_A + 2R_B)C_1$, so the measured time is linearly dependent on the plant's resistance (R_B). I made a few quick measurements of plants; this was not a very rigorous or scientific investigation, but it indicated that R_B might range from 300kΩ to 30MΩ although the true range is probably wider.

The exact limits are not as important as the fact that the range covers at least two orders of magnitude and goes up to very large resistance values, in the of tens of megohms.

The MIDI Sprout schematic gives C_1 = 4.2nF, which is an unusual value. However, the bill of materials lists the much more common 4.7nF. The difference is not significant, given the large resistor R_B range. Using 4.7nF with the assumed range of R_B and R_A = 100kΩ gives oscillation periods from about 2.2ms to just under 200ms.

The software takes multiple consecutive samples of T (and hence plant resistance) and performs some statistics on the data. It maps the results to MIDI musical notes (there are 127 possible note values).

It also generates some MIDI control commands, which can be used to vary some qualities of the sound generated, depending on the configuration and capabilities of the synthesiser or DAW (digital audio workstation) software receiving the MIDI commands.

The note-generation algorithm can potentially produce new note values at rates that are either too slow or too fast to be useful in a given situation. Therefore, an input parameter to the algorithm is used to adjust its behaviour to suit the taste of the user, with the value obtained via a potentiometer connected to an ADC input of the microcontroller (see Fig.7).

The note values generated depend on the fluctuations in the plant's resistance; they are not a simple linear map of resistance to note. The latter would not work because different plants have significantly different resistances (and it depends on exactly where the probes are connected), so a given connection to a plant would cover too limited a range.

Once a reliable connection is made, the note generation algorithm used will respond to small variations in resistance due to biological processes in the plants (eg, water transpiration), physical movements or human touch, resulting in a varied musical output.

The MIDI Sprout has several LEDs that provide a light show to indicate that plant resistance is being processed to generate notes. The LEDs can also be used in conjunction with the potentiometer and a switch (not shown) to implement a simple menu system for device configuration. This was included in later versions.

Potential-divider measurement

As mentioned above, resistance can also be measured by a microcontroller using a potential divider (connected to an on-chip ADC), as in Fig.8. One of the resistors is the unknown resistance being measured (R_x) and the other is a known fixed resistor (R_1). Fig.8 shows the resistor being measured as the grounded resistor, but it could it be configured the other way around.

The potential divider approach to resistance measurement has the advantage of a low component count and being simple to implement, but there are some problems.

The well-known potential divider formula gives the voltage measured by the ADC as $v_m = R_x V_S \div (R_1 + R_x)$. This can be rearranged to find the resistance being measured as:

$$R_x = \frac{v_m R_1}{(V_S - v_m)}$$

This requires a little more calculation in the microcontroller code than the 555-based measurement, but the calculation is not too onerous.

The measurement depends on the voltage applied to the potential divider, which in the simplest case is the supply voltage (V_S in Fig.8). This makes it seem like the measurement would be sensitive to supply voltage changes, unlike the timing of the 555 astable (as discussed above).

However, most microcontroller ADCs can use the supply voltage as the reference. In this case, any effect of varying supply voltage on the measurement is cancelled out by the fact that the

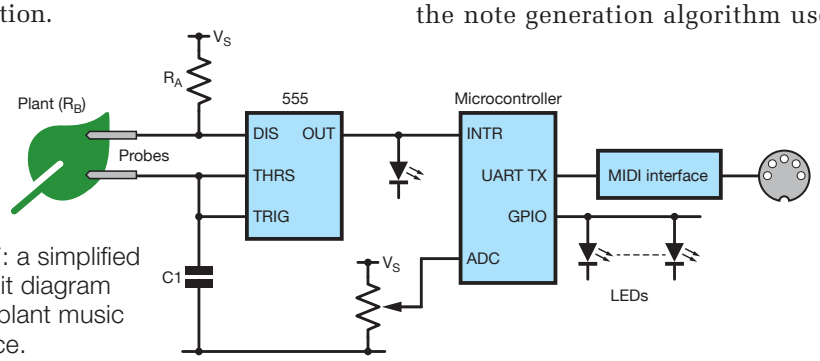


Fig.7: a simplified circuit diagram of a plant music device.

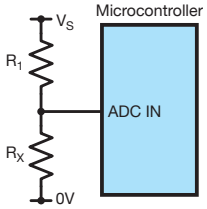


Fig.8: making resistance measurements using a potential divider.

reference voltage is varying in exactly the same way.

In our discussions, we will use V_S = 5V for simplicity. The Midi Sprout uses three AA cells, so V_S will likely range from 4.5V down to 3.6V, within the ATmega328P's 2.7-5.5V operating range. The specific value of V_S is not a critical factor in the discussions below.

We will assume we want to measure the resistance range discussed above (300kΩ to 30MΩ). The optimal value for the fixed resistor (R_1) is the geometric mean of the upper ($R_{x,max}$) and lower ($R_{x,min}$) values of the required measurement range, that is:

$$R_1 = \sqrt{R_{x,min} \times R_{x,max}}$$

For our example, we get R_1 = 3MΩ. When R_x=R_1, the measured voltage will be half the voltage applied to the potential divider.

The large resistance values in our example may lead to specific problems, which we will discuss shortly, but we will start with aspects of the behaviour of potential divider resistance measurement that do not depend on where the range to measure is centred; the same concerns would apply to a measurement range of 300Ω to 30KΩ, assuming the fixed resistor (R_1) is scaled accordingly.

A key problem with this approach is that the relationship between v_m and R_x is non-linear. The form of the curve depends on whether R_x is the upper or lower resistor in the potential divider. Fig.9 shows the measured voltage against R_x, when R_x is the lower resistor, and Fig.10 is for R_x as the upper resistor. These plots are for the values of R_1 and V_S and the R_x range discussed above.

The curve is very flat for larger resistance values; it takes a very large resistance values to reach the maximum (Fig.9) or minimum (Fig.10) ADC code. Thus, this approach does not realistically use the full range of the ADC. For an N-bit ADC, the range of (non-zero) voltages the ADC can measure is from $v_m = V_S \div 2^N$ to $v_m = (2^N\text{-}1)\,V_S \div 2^N$.

With R_1 = 3MΩ and a 10-bit ADC (as on ATmega328P used in the MIDI Sprout), this corresponds to a measured resistance range from 2.93kΩ to 3069MΩ, much larger than the required range discussed above. The larger values are unlikely to be realistically measurable this way, but here we are interested in the general behaviour of the potential divider.

With a lower centre resistance, a relatively larger range would be feasible, but the flat part of the curve always means relatively large resistance differences between ADC codes, which may be a problem.

Using a larger value for R_1 results in a more linear response, but causes a given resistance measurement range to map onto a smaller measured voltage (ADC code) range. Fig.11 shows the response with the same parameters as Fig.9 except R_1 = 20MΩ. This value may be impractical, but the aim is to show the relative response with lager R_1, and this applies for lower resistance ranges.

A 10-bit ADC's output code for the measurement (M) is given by $M = 1024\,V_m \div V_{ref}$, where V_{ref} is the ADC's reference voltage, which is typically the supply voltage (V_{ref}=V_S). Substituting the potential divider equation into the code equation gives:

$$M = \frac{1024\,R_x}{\left(R_1 + R_x\right)}$$

Using the minimum and maximum values for R_x gives us the minimum (M_{min}) and maximum (M_{max}) ADC codes for a required measurement range. The difference between these limits tells us how many ADC codes are actually used, and 1024 ÷ (M_{max} – M_{min}) is the proportion of possible ADC codes used. We can use 2^N for different N-bit ADCs instead of 1024 (2^{10}).

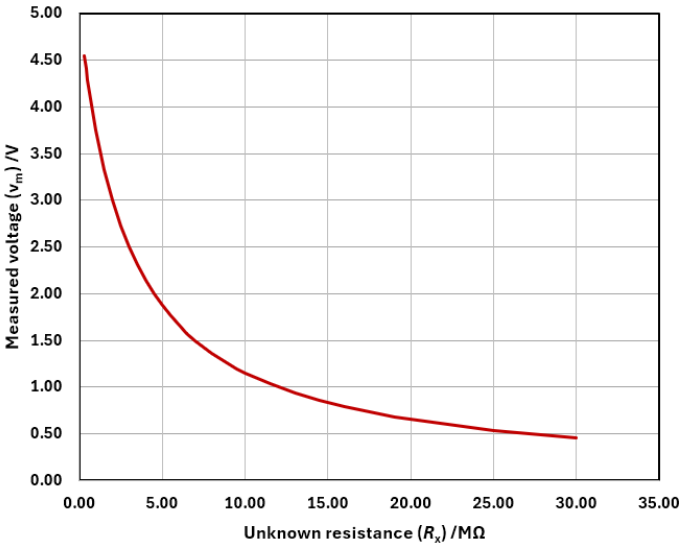


Fig.9: potential divider voltage vs lower resistance.

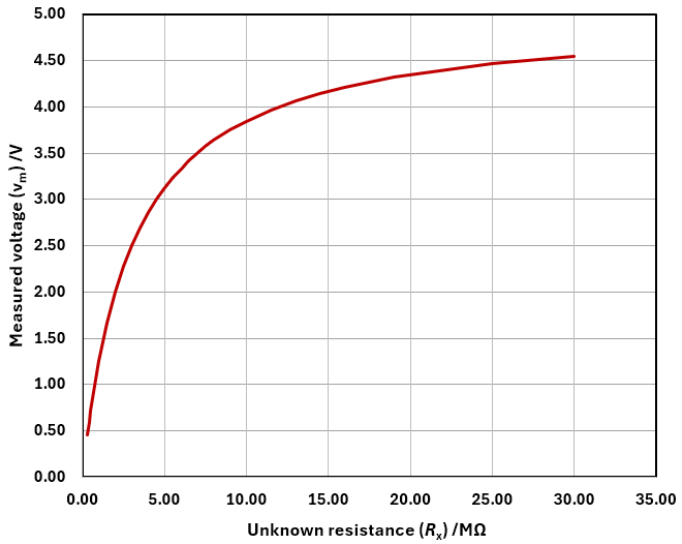


Fig.10: potential divider voltage vs upper resistance.

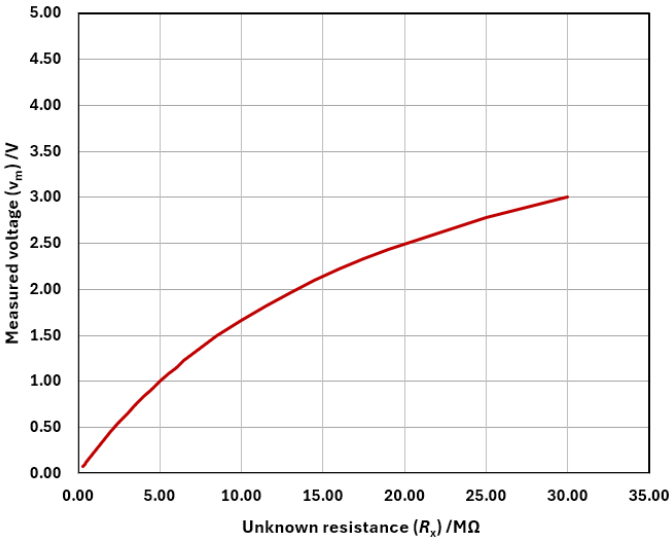


Fig.11: increasing the fixed resistor value provides a more linear response but a smaller ADC (Vm) range.

Simulation files

Most *Circuit Surgery* columns feature the use of the free circuit simulation software LTSpice. It is used to support descriptions and analysis in *Circuit Surgery*.

The examples and files for this issue are available for download: https://pemag.au/link/acby

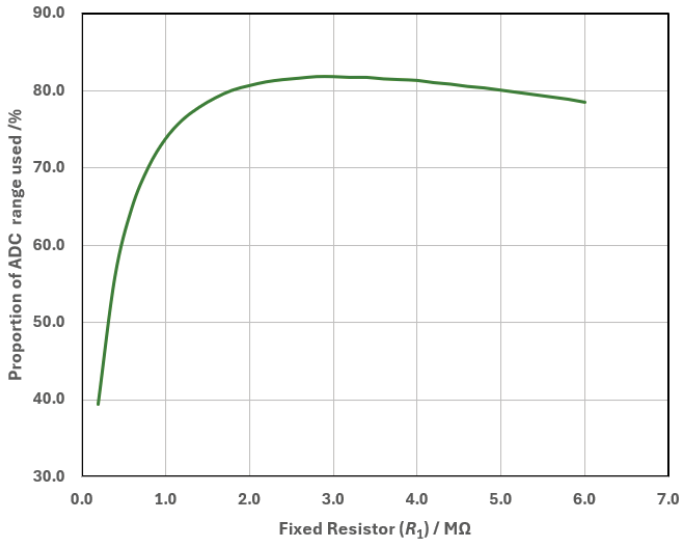


Fig.12: the proportion of ADC codes used in potential divider resistance measurement vs the fixed resistor value.

Fig.12 is a plot of how the proportion of possible ADC codes used varies with R_x for the 300kΩ to 30MΩ example measurement range. There is a peak (838 codes, 82% of ADC range) at R_x = 3MΩ, which is the geometric mean of the range limits mentioned earlier.

Source resistance

We mentioned a couple of times that the resistance values involved in the example measurement are very high. This is due to the specific application we are considering and would not apply in all cases. However, the source resistance of the potential divider must be considered whenever this technique is used.

Circuit outputs can be modelled as a voltage source in series with a source resistance (R_s); this is called a Thévenin-equivalent circuit. For a potential divider, the voltage source value is equal to the unloaded potential voltage (the output with nothing connected to it). The source resistance is equal to the parallel value of the two resistors; for our circuit, that is $R_s = R_1R_x \div (R_1 + R_x)$.

Over our range of R_x values, R_s varies from 273kΩ to 2.73MΩ.

Fig.13 shows a simplified equivalent circuit for a potential divider connected to an ADC input. The ADC is modelled as an input resistance (R_{in}), sampling switch, and sample-and-hold capacitor (C_{SH}). A slightly more detailed model of the ATmega328P's ADC input circuit is shown in Fig.14.

Before the ADC makes a conversion, it closes the sample switch to charge the sample and hold capacitor (C_{SH}) to the input voltage (v_{in}). For accurate measurement of R_x, C_{SH} must charge to v_{in} within one least significant bit (LSB) of the ADC.

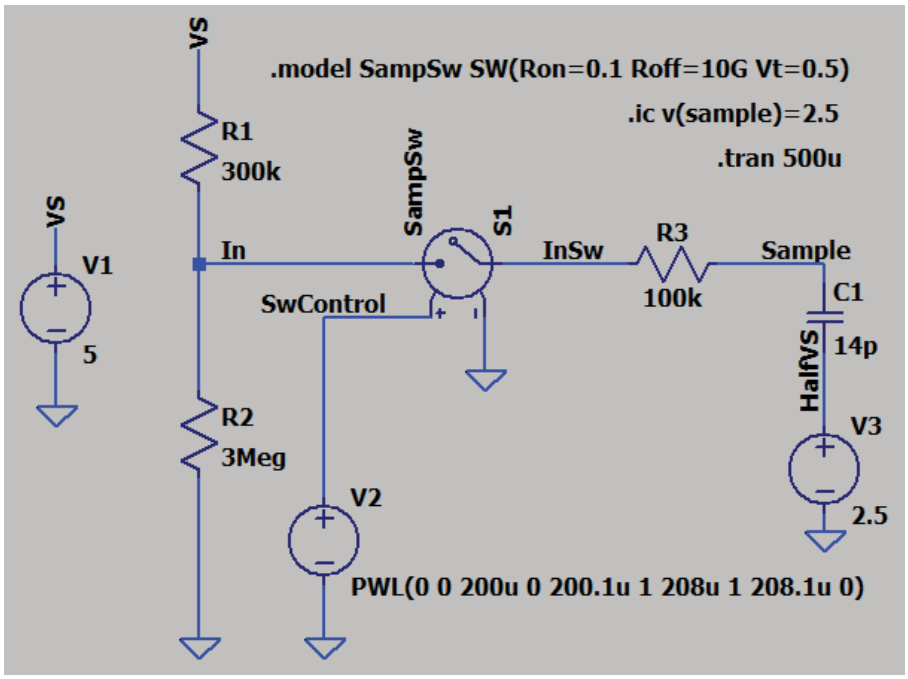


Fig.15: an LTspice model for ADC sampling via a potential divider.

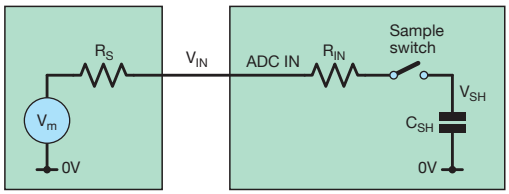
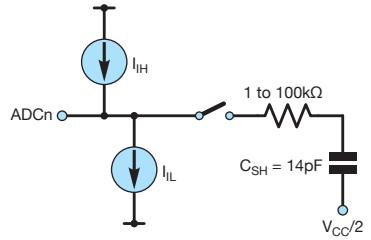


Fig.13: a simplified circuit for the potential divider connected to an ADC input.

Fig.14: the ATmega328P ADC input circuit model (from its data sheet).



The LSB is the voltage difference between two adjacent codes in the ADC's output. For an N-bit ADC, this is a fraction of $1 \div 2^N$ of the conversion range.

If the source resistance is large, the time taken to charge the sampling capacitor may be excessively long, which may result in inaccurate measurements because the ADC converts before the sampling capacitor has been fully charged.

The time taken for an RC circuit to charge to a fraction (F) of the applied voltage is given by $t = -RC\log_e(1 - F)$. The worst case is when the ADC is charging from 0 to the maximum code value ($2^N - 1$).

For a 10-bit ADC, this is 99.9% ($F = 1023 \div 1024 = 0.999$) of the voltage applied to the potential divider, assuming it is equal to the ADC full-scale voltage (the reference voltage, which corresponds to 2^N).

This gives us $t = -RC\log_e(1 - F) \gg 7RC$. For C = 14pF and R = 2.73MΩ (the worst case for our example), we have 268µs.

The ATmega328P data sheet indicates a default ADC clock of 50-200kHz and that the sample-and-hold completes 1.5 cycles after conversion is initiated.

This implies 7.5-30µs for the sampling capacitor to settle, which is significantly shorter than RC charge time just calculated, indicating that the conversion would not work correctly in this case.

Fig.15 shows an LTspice schematic of a potential divider and ADC input model similar to Figs.13 & Fig.14. In this example, the sampling switch is closed for 8µs.

With the component values as shown in Fig.15, the resulting waveforms are shown in Fig.16. This illustrates a failure of ADC sampling due to the high impedance of the potential divider.

The ATmega328P data sheet recommends source impedances less than 10kΩ for driving the ADC to avoid this problem.

The lower pane of Fig.16 shows the sample switch control. The switch (S1) is on from 200µs to 208µs, during which time the sampling capacitor (C1) should charge to the input voltage, V(in), which is the voltage produced by the R1/R2 potential divider.

When the sampling switch is closed, the voltage on the sampling capacitor – V(sample), the yellow trace in the upper pane – starts charging towards V(in), shown by the green trace. However, due to the high source impedance of the potential divider, it has not reached the input voltage by the end of the sampling period – the sampled voltage is incorrect.

Also, V(in) dips significantly due to the loading effect of the ADC input, although it would recover given time.

Fig.17 shows the results of the same circuit with the potential divider resistors changed to 1kΩ and 10kΩ (everything else is the same). Here, the sample voltage is able to reach the input voltage during the 200-208µs sampling period, so the ADC will obtain a correct measurement. Also, the dip in the input voltage due to loading is much smaller.

We conclude that it is not practical to connect a high-resistance potential divider directly to the ADC. A solution is to use an op amp buffer as shown in Fig.18. The op amp needs high input impedance, low input bias current and rail-to rail operation (the full range of V_{in} as shown in Fig.9 is within 0.5V of both supplies).

Another possible solution is to add a capacitor from the ADC input pin to ground. That will form an RC low-pass filter with the source impedance, slowing the system's response, but it will also provide a local charge reservoir for the ADC input during sampling.

Many PICs and other microcontrollers allow you to specify an extended sampling time by programming the appropriate registers, allowing higher source impedances at the cost of slowing down the ADC process. Unfortunately, the ATmega328P doesn't allow this, but you could run multiple ADC conversions and discard all but the final result, giving the sampling capacitor additional opportunities to settle.

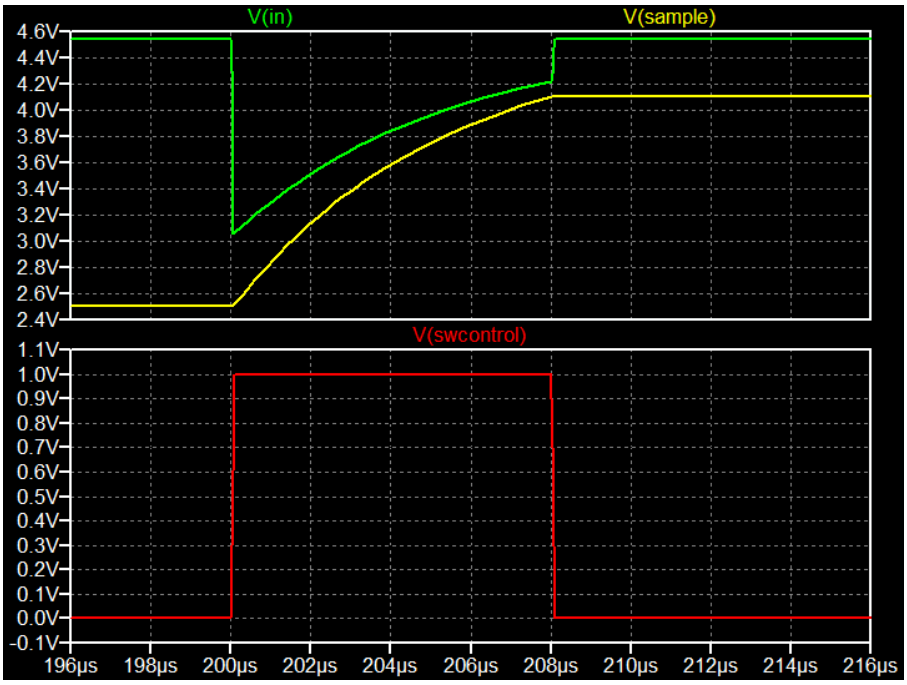


Fig.16: the simulation results for the circuit in Fig.15.

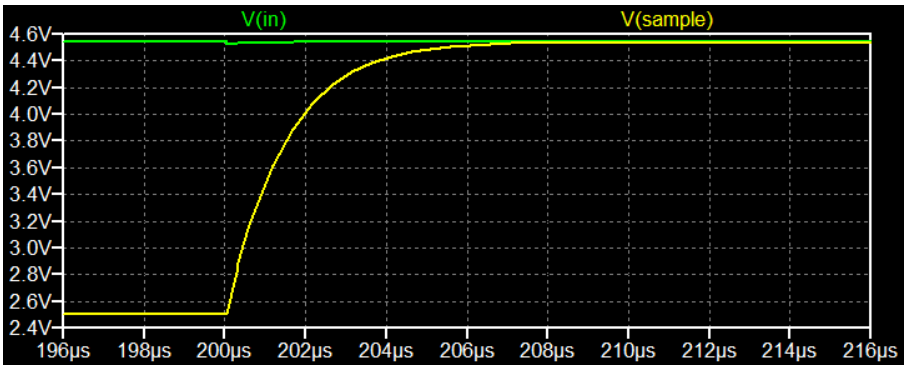


Fig.17: the simulation results for the circuit in Fig.15 with R1=1kΩ & R2=101kΩ.

Further comparisons

The resistive potential divider (resistance-to-voltage) and 555 timer (resistance-to-time) are both useful low-cost techniques for resistance measurement using microcontrollers.

We have described these approaches in the context of devices that have to measure resistances up to tens of megohms and where good resolution and sensitivity to variations are useful, but accurate measurement of the actual resistance is not required.

We assumed a specific range of resistances for this comparison, which may not be a perfect match to the values encountered by a plant music device, but is sufficient to allow discussion on the merits of the two techniques in this application.

The requirement to measure high resistance means that a buffer would be needed for the potential divider (as in Fig.18). This adds to the component count and cost, but the component count is still smaller than the 555 circuit.

A key advantage of the potential divider circuit is its simplicity. If we assume the microcontroller can measure time to 1µs resolution (using a microsecond time function call; more accurate measurements are possible)

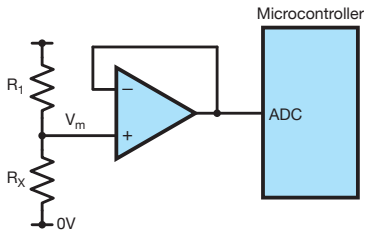


Fig.18: making a resistance measurement using a buffered potential divider.

and the 555 astable with the resistance range specified has a period ranging from approximately 2.3ms to 197ms, this equivalent to 74% of an 18-bit ADC's range.

The bit-equivalence is found using $\log_{10}(range) \div \log_{10}(2) = \log_{10}(197000 - 2300) \div \log_{10}(2) = 17.6$, so 18 bits (round up) are required to cover the number of steps from 2300 to 197,000.

The potential divider method on the same microcontroller uses 82% of its 10-bit ADC range. This is significantly less than the timer-based approach. Even if the resolution of the timer approach is 10µs (rather than 1µs), the equivalent resolution (15 bits) is still significantly higher.

Equivalent (or effective) number of bits is a useful way to compare the resolutions of systems, and in more advanced analyses than we have done here can account for factors such as electrical noise. However, if this seems a bit abstract, we can simply directly compare the around 830 possible measurement values for the potential divider with 190,000 for the timer.

The poor resolution of the potential divider is exacerbated by the non-linearity of its response. This means poor resolution over a significant part of the measurement range, which is not a good fit for the requirements of the plant music device, which needs to measure resistance fluctuations that could be relatively small.

Note that using a potentiometer (as in Fig.7) to provide an input to the ADC does not suffer from non-linearity (ignoring any imperfections in the potentiometer).

For a linear potentiometer, the ADC reading will be proportional to the amount of rotation. The non-linearity in the resistance measurement scenario is due to the fact that the total resistance changes as the sensor (or plant in our example) resistance changes.

The timer in this example takes much longer to make a measurement than the ADC connected to the potential divider. This does not matter because music note generation only has to run at a rate typical for music and the timing is sufficient for this. Music is event-based; that is, notes going on and off, and other changes occur in a time sequence.

Using a timer trigger dependent on the plant's resistance is perhaps a more natural implementation than the microcontroller deciding when to generate notes and then making a measurement.

Despite not being the best approach for the plant music device, potential-divider-based resistance measurements are suitable for other resistance measurement applications, particularly for resistive sensors such as light-dependent resistors, thermistors and force/flex sensors. It is an effective solution when low-to-moderate accuracy is required in a low-cost system.

Conclusion

We have looked at an open-source design and considered a possible alternative approach. A case study like this can be a useful way of learning about electronics. Hopefully, the original designer made good choices, and you will be able to see why that was so (at least at the time it was created). If not, then you may be able to find a better implementation. PE

5-year collections

2019-2023
£49.95

2018-2022
£49.95

2017-2021
£49.95

2016-2020
£44.95

Purchase and download at:
www.electronpublishing.com

ETI BUNDLE (1) Teach-In 3, 4 and 5 – all on CD-ROM – only £18.95

ELECTRONICS TEACH-IN 3 – CD-ROM
Mike & Richard Tooley

The three sections of the *Teach-In 3* CD-ROM cover a huge range of subjects that will interest everyone involved in electronics – from newcomers to the hobby and students to experienced constructors and professionals.

The first section (80 pages) is dedicated to *Circuit Surgery*, *EPE/PE's* regular clinic dealing with readers' queries on circuit design problems – from voltage regulation to using SPICE circuit simulation software.

The second section – *Practically Speaking* – covers hands-on aspects of electronics construction. Again, a whole range of subjects, from soldering to avoiding problems with static electricity and identifying components is covered. Finally, our collection of *Ingenuity Unlimited* circuits provides over 40 circuit designs submitted by readers.

The CD-ROM also contains the complete *Electronics Teach-In 1* book, which provides a broad-based introduction to electronics in PDF form, plus interactive quizzes to test your knowledge and TINA circuit simulation software (a limited version – plus a specially written TINA Tutorial).

The *Teach-In 1* series covers everything from electric current through to microprocessors and microcontrollers, and each part includes demonstration circuits to build on breadboards or to simulate on your PC.

ELECTRONICS TEACH-IN 4 – CD-ROM
A BROAD-BASED INTRODUCTION TO ELECTRONICS
Mike & Richard Tooley

The *Teach-In 4* CD-ROM covers three of the most important electronics units that are currently studied in many schools and colleges. These include, Edexcel BTEC level 2 awards and the electronics units of the Diploma in Engineering, Level 2.

The CD-ROM also contains the full *Modern Electronics Manual*, worth £29.95. **The Manual** contains over 800 pages of electronics theory, projects, data, assembly instructions and web links.

A package of exceptional value that will appeal to anyone interested in learning about electronics – hobbyists, students or professionals.

ELECTRONICS TEACH-IN 5 – CD-ROM
JUMP START
Mike & Richard Tooley

15 design and build circuit projects for newcomers or those following courses in school and colleges.

The projects are: ■ Moisture Detector ■ Quiz Machine ■ Battery Voltage Checker ■ Solar-Powered Charger ■ Versatile Theft Alarm ■ Spooky Circuits ■ Frost Alarm ■ Mini Christmas Lights ■ iPod Speaker ■ Logic Probe ■ DC Motor Controller ■ Egg Timer ■ Signal Injector Probe ■ Simple Radio Receiver ■ Temperature Alarm.

PLUS

PIC'n' Mix – starting out with the popular range of PIC microcontrollers and *Practically Speaking* – tips and techniques for project construction.

The CD-ROM also contains:
■ Complete *Teach-In 2* book, a practical introduction to PIC microprocessors
■ MikroElektronika, Microchip and L-Tek PoScope software.

Three Teach-ins for the great price of £18.95

PLUS you also get the contents of the free CD-ROM from each issue...

... so that's another TWO Teach-Ins and The Full Modern Electronics Manual!

What a Bargain!!

Looking back and ahead

Techno Talk

From clattering split-flap displays and glowing VFDs to holographic smart glasses, liquid-cooled smartphones, and virtual acoustic worlds; what might tomorrow bring?

Max the Magnificent

As usual, I find what I laughingly call my mind bouncing back and forth between the technologies that have been and gone, those that are with us today, and those that have yet to grace us with their presence.

Displays of yesteryear

When creating embedded systems, test equipment, IoT devices, or industrial control systems, we're spoiled for choice when it comes to the smorgasbord of display technologies that are available now, including LCD, LED and OLED screens; even humble 7-segment LED displays. It wasn't always this way.

My current hobby project, which I'm documenting in my *Cool Beans* column, features vacuum fluorescent displays (VFDs). Working with these little scamps has prompted me to ponder what came before.

Displays that go clunk in the night

Before electronic displays existed, engineers relied on ingenious electromechanical solutions. Needles, gears, drums, shutters, flaps, rotating prisms along with countless other moving parts were pressed into service to present information.

Some displays showed a single measurement, others displayed numbers or text, and still others created entire images using arrays of mechanical pixels. Many of these devices were bistable, requiring power only while changing state—an idea that remains attractive even today.

I'm thinking about analog meters, split-flap displays, flip-dot displays, rotating-prism displays and a host of other ingenious electromechanical contrivances that engineers devised before electronic displays became practical.

A split-flap display in Frankfurt. Source: Ratopi (https://w.wiki/Ruer).

I was born and bred in Sheffield, Yorkshire. When I was a little lad in the early 1960s, my dad used to take me into the city centre on Saturday mornings to give my mum time to clean the house (I still have no idea how the two of them managed to make so much mess!). We'd usually catch the bus there. On a good day, we'd catch the train home.

Panels of split-flap displays hanging above the platforms presented a constant stream of information regarding arrivals, departures and delays. I can still remember the rolling roar of those displays updating as though it were yesterday.

As with many things, retro technologies like split-flap displays are making a comeback. One company I really like is Oat Foundry (www.oatfoundry.com), which makes a modular, modern incarnation of split-flap displays that leaves me drooling with desire.

If you want to know more about how these bodacious beauties work, or even build your own, check out Scott Bezek's YouTube video at youtu.be/UAQJJAQSg_g

From filaments to phosphors

As clever as these electromechanical displays were, they all suffered from the same fundamental drawback: moving parts. Gears wore out, springs weakened, bearings became sloppy, and everything made at least some noise. Engineers naturally began looking for ways to create displays using light rather than motion.

Some of the earliest segmented displays employed small incandescent lamps, with one bulb illuminating each segment. Gas-discharge technologies soon followed, including Panaplex displays and the much-loved Nixie tube. Each represented a step toward displays that were brighter, more reliable and easier to control electronically.

Then came vacuum fluorescent displays (VFDs), which combined the unmistakable charm of vacuum-tube technology with the flexibility of segmented electronic displays. By the 1970s and 1980s, these glowing little beauties were everywhere; from calculators, microwave ovens, and VCRs to hifi systems, laboratory instruments and vehicle dashboards.

Which brings us back to my own project. In next month's column, I'll be firing my VFD tubes up for the first time. I cannot wait to feast my orbs on their glorious, retro-futuristic blue-green fluorescent glow.

Displays that disappear

As fascinating as vintage display technologies may be, I'm even more excited by what comes next. I sometimes wonder whether future displays will disappear altogether. By this, I mean that displays will become personal rather than public. Instead of presenting information for everyone to see, they will present it only to the person for whom it is intended, unless that person chooses to share it.

Imagine, for example, wearing an ordinary-looking pair of glasses. Tiny cameras see everything you see. The glasses communicate with an AI running locally on your smartphone via Bluetooth, while also reaching out to more powerful cloud-based AI whenever necessary.

The system remembers where you left your keys, recognises every bird, flower, building and gadget you happen to glance at, answers questions about whatever has caught your attention, and overlays the real world with useful information.

Looking for the nearest café? Glowing arrows appear floating on the pavement in front of you, together with the distance and estimated walking/arrival times. Need to repair a piece of equipment? Step-by-step instructions appear to hover over the real hardware.

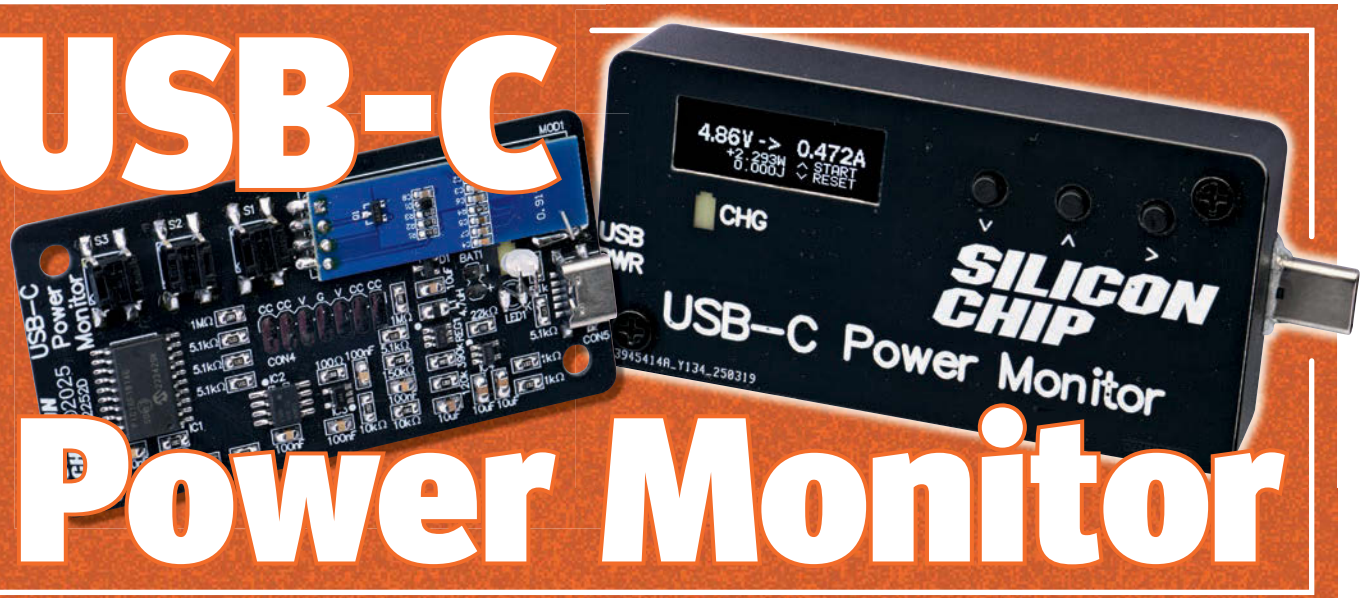
Part of the challenge is in the display. Existing AR headsets rely on waveguides and complex optics that tend to be bulky, expensive and power-hungry. What we really need are lightweight spectacles that people (by which I mean me) would happily wear all day.

One technology that recently caught my eye (no pun intended) comes from a company called Swave Photonics.

continued on page 75...

Part 2 by Tim Blythman

USB-C Power Monitor



This compact device lets you monitor the voltage, current, power and energy supplied to a USB-C device. The first article last month covered some background information, the circuit details and reasons for the design choices. Now we'll describe the construction and usage of this handy tool.

The USB-C Power Monitor can measure the V_{BUS} voltage and current flowing between two devices; using this information, it can also calculate power and energy. That makes it similar to our USB Power Monitor that was published in the August 2016 issue, but with USB-C connectors, a more comprehensive display and extra capabilities.

Since USB-C allows current to flow in either direction, the new Monitor must handle that, as well as USB-C's higher voltage and current capabilities, up to 48V and 5A. It has an internal rechargeable battery to avoid loading the USB host.

The Monitor also tests the state of the configuration channel (CC) lines, which are also new to USB-C. It has an OLED display module and three tactile switches for control. All these features are packed into a compact 80mm × 40mm enclosure.

Construction

The two PCBs are connected by soldered wire, ribbon cable or FFC (flat flexible cable) connections. That's because pluggable connections have variable resistance and will interfere with the correct operation of the current shunt monitor. The smaller PCB has the USB-C plug and socket. We will build this first, since it is easy to test.

The second PCB carries most of the parts and also forms the front panel of the completed unit. This PCB can operate by itself, without a battery, so the second PCB can also be tested for basic functionality before everything is joined.

The case we have chosen requires three cutouts to be made. These are not too tricky, and they can also be tested for fit before the final assembly step. As you can see from our photos, the completed unit is compact and neat.

Both PCBs are 0.8mm thick and feature surface-mounting parts, so you will need the usual SMT gear; a fine-tipped soldering iron (or medium/chisel tip, if you prefer), solder, flux paste, tweezers, a magnifier and good lighting. Solder wicking braid will come in handy, too. Work outside or with good ventilation to avoid inhaling smoke from the flux.

Connector PCB

You might not need them, but Fig.4 shows the overlay diagrams for this PCB, which is coded 04102251 and measures 78 × 11mm (it's 0.8mm thick). There aren't many parts on it, but we think it's the most tricky to solder because of the fully featured USB-C connectors and their fine pin pitch. We found socket CON2 to be most challenging, so we recommend starting with that.

We haven't tried it, but if you have a hot-air station and solder paste and are familiar with using them, then you might like to use them to assemble the connector PCB. This process would be closer to the reflow process used for commercial soldering of these parts.

CON2 is much the same part that we used in the USB Cable Tester that was published in the November & December 2022 issues, although this time we are using a variant with shorter through-hole pins since the PCB is thinner. These pins are very fine and can easily bend if they are bumped; this could lead to short circuits with other pins, so be gentle.

Place CON2 on the PCB and tack one of the larger shell pins, then confirm that the SMT pins on the top of the PCB are aligned and flat against the PCB. This should avoid the possibility

Silicon Chip Short-form Kit (SC7489, ~£38 + P&P): *this kit includes all the non-optional parts listed except the case, lithium-ion cell and glue. It does include the FFC (flat flexible cable PCB) for joining the two PCBs (see p78).*

USB-C Power Monitor, part two

Fig.4: there aren't many components on this PCB; it does little more than pass all USB-C signals through and break out a few of them for connection to the main PCB. Soldering CON1 and CON2 will probably be the trickiest part of this project.

of the part moving while you are trying to solder it. Clean the iron and add a small amount of fresh solder.

Carefully add flux to the smaller pins and solder them. We've extended these pads where possible, as this will allow you to touch the iron to the pad and see that the solder has flowed and melted onto the pin. Flip the PCB over and solder the through-hole pins, including the four larger shell pins.

If you think you have a bridge between pins, carefully use the braid and a little more flux paste to draw the excess solder away. The solder's surface tension should hold enough solder behind to make a solid joint. If in doubt, add a little more, with some more flux to ensure it flows smoothly.

The USB-C plug (CON1) is a straddle-mount part that clips over the edge of the PCB. This helps to lock it into place so it's not likely to move during soldering. Its pins are also fine, and if they are bent at all, particularly sideways, then they may bridge to other pins. If it has a protective cap, leave it in place for now.

Place the plug over the edge of the PCB and check that it is flat against the PCB and all the pins on both sides align with their pads. Carefully slide it sideways if needed. It should be parallel to the edge of the PCB, too.

Add flux to all the pins and touch the iron to the end of each pad in turn until the solder flows onto the respective pin. The shell can also be soldered to the larger pads on the outside edge of the PCB; this will add mechanical strength. Check for solder bridges and use the braid with extra flux paste as needed.

Once you are happy with it, fit the three resistors to this PCB. The larger part is the 15mΩ shunt; the other two are 220Ω. Clean up the excess flux on the PCB using the solvent of your choice. Some fluxes recommend specific chemicals, but we find that isopropyl alcohol is a good alternative.

Allow this PCB to dry and give it another inspection. If there is a short circuit between any pins, it will not work as designed. Remove the cap on CON1. At this stage, this PCB should be functional as a USB-C extension, so you can test it by connecting a fully featured USB-C cable (with plug) to CON2. Be sure to rotate the connector 180° to test both configurations.

Ideally, our USB Cable Tester should be used for initial tests, since this will not cause or suffer any damage if there is a fault. The extended cable should give exactly the same results as the cable on its own, since all the lines are taken straight through. There may be a slightly higher cable resistance due to the extra cable junctions.

Lacking that, connect a USB-C device you don't care about too much to a computer through this extension and check that it works normally.

We also used a USB-PD power supply connected to a USB-PD trigger device to test this PCB, since they are fairly cheap and will exercise the power and CC (configuration channel) wires. See the end of this article for an example of the boards we used for testing.

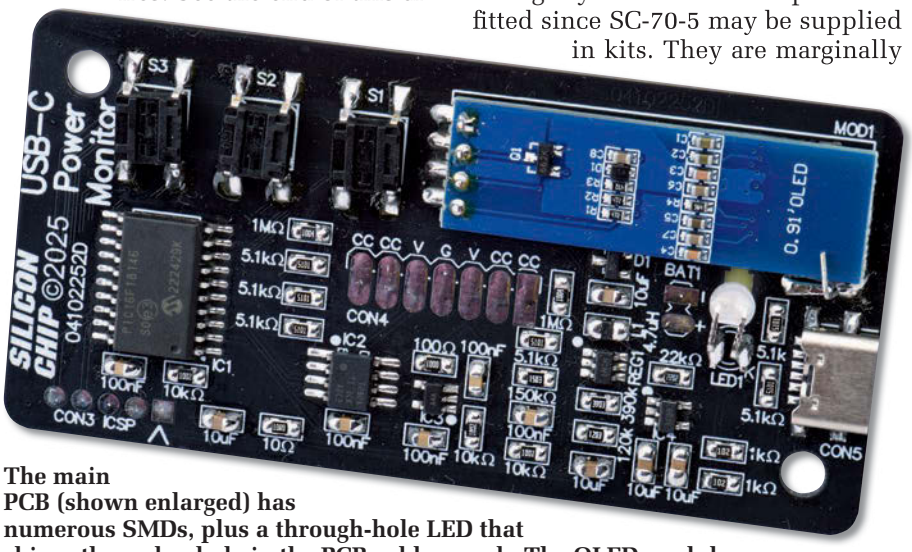
If the tests work as expected, then test out the data wires using a computer or similar. Be sure to use it with a USB 3.x device so that you know that all the USB data circuits are functional. As before, the extended cable should give the same results as the cable on its own. If you run into any problems, go back and rectify them before using this PCB.

Main PCB

While the second PCB, coded 04102252 and measuring 80 × 40mm, has a few smaller parts, in comparison, it should be relatively straightforward (see Fig.5).

Start with the smaller ICs and the regulator. IC3 is the AD8541 op amp (or similar) and should be labelled with the code A4A or A12, while IC4 is the MCP73831 charge controller and should be labelled KD, followed by a two-character lot code. They are both in SOT-23-5 (five lead) packages.

Note that the pads for IC3 have been modified to allow either an SOT-23-5 or slightly smaller SC-70-5 part to be fitted since SC-70-5 may be supplied in kits. They are marginally



The main PCB (shown enlarged) has numerous SMDs, plus a through-hole LED that shines through a hole in the PCB solder mask. The OLED module is also mounted to be visible through a cutout in the PCB. A row of header pins can be fitted to CON3 if in-circuit programming of IC1 is required.

smaller, so not too much more difficult to solder. If you're sourcing the parts yourself and prefer to use an SOT-23-5 part, it should fit the same pads just fine (it's a dual footprint).

REG1, the MCP16252, is similarly small in a six-lead SOT-23-6 package and marked with the code MC and a two-character lot code. Be sure to check the orientation on this one, since it is symmetrical, unlike the five-lead parts. The alignment dot is below and to the left of the MC code; in other words, pin 1 is at lower left when the text is upright.

Double-check its orientation before soldering more than one pin as it's tricky to fix if you get it wrong!

Follow with IC1 and IC2, the two SOIC parts; these should be a breeze after the smaller parts, but you still need to pay careful attention to their orientations.

Diode D1 is quite small, but should be easy enough to manage. CON5, the third USB-C connector should be aligned flush with the edge of the PCB as much as possible.

After the resistors, capacitors and sole inductor are fitted, this PCB will be fairly complete. There are only two different capacitor values; the five 10µF parts might be thicker than the five 100nF parts. They will be unmarked, though, so be sure not to mix them up.

There are several different resistor values, so match the markings to the silkscreen or use a multimeter if you are unsure. On the PCB, the silkscreen values are generally below or to the right of the part when they are in a row, to avoid ambiguity. You can also check against the overlay diagram (Fig.5) and photos.

You can also solder in the 3mm LED now. We're using a through-hole part here, since two-lead SMD bicolour LEDs are not widely available. The K marking on the PCB corresponds to the cathode of the green LED in the package. Bend the leads as shown in the photos and solder it to the PCB. Keep the lead offcuts for later.

Now is a good time to clean up the flux residue on the PCB, before the OLED and tactile pushbuttons are fitted; these parts will not appreciate being immersed in solvent. Allow it to dry and inspect the soldering for any poor connections or bridges and rectify them if needed. Fig.6 shows the state of the PCB at this point.

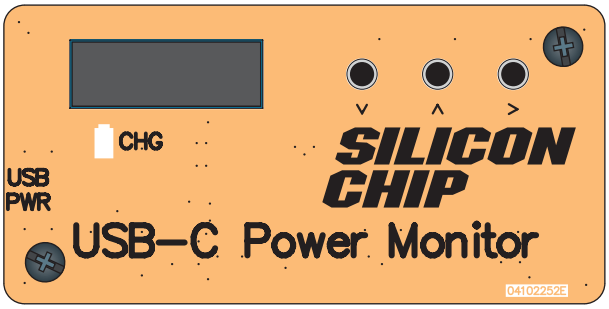
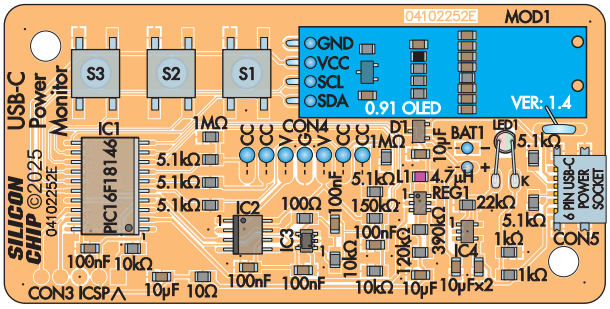


Fig.5: this PCB hosts most of the parts. Don't mix up the various SOT-23/SC-70 parts (in 3-pin, 5-pin & 6-pin variants). Fortunately, there is a fair amount of space on the PCB, so the silkscreen markings should be easy to follow.



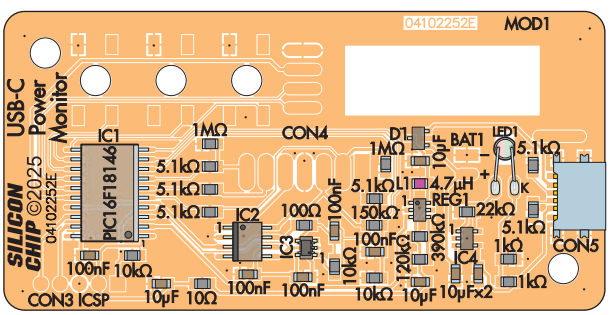


Fig.6: the partially assembled main PCB – you can test-fit the boards before they are turned into a single assembly. We recommend you take your time and ensure that they are a good fit individually before joining them together.

Finishing the main PCB

You can now fit the OLED module and tactile switches to the main PCB. For the OLED module, desolder any header that has been attached to it and clean up the four pads.

The easiest way to do this is to add solder to the pins so that they are all covered in a single, large blob of solder and then heat that while pulling the pins with pliers until they slide out of the board. After that, use a solder sucker to remove the excess solder bridging the pads.

Now solder short wire offcuts to the four pads on the main PCB. These should point straight up and align with the holes in the OLED when it is overlaid. The best way to do this is to bend short pieces of wire into an 'L' shape and use the extended pad to secure them better. It's a bit fiddly, but we think it leads to a very tidy result.

Remove the OLED's protective film and slot the OLED over the offcuts, then solder the OLED module to them. Be sure that the OLED is not too close to the edge of the PCB; otherwise, it could foul the enclosure. Another piece of wire can be soldered to the long pad near LED1 and bent over the OLED module. This will provide support to the other end of the OLED.

Fit the pushbutton switches next. Their stems poke through the PCB to face outwards, so you may need to sit the PCB up using spacers or a PCB holder. Tack one lead and adjust the switches so that their stems are centred in the holes. This will look better and eliminate the chance of the switches binding.

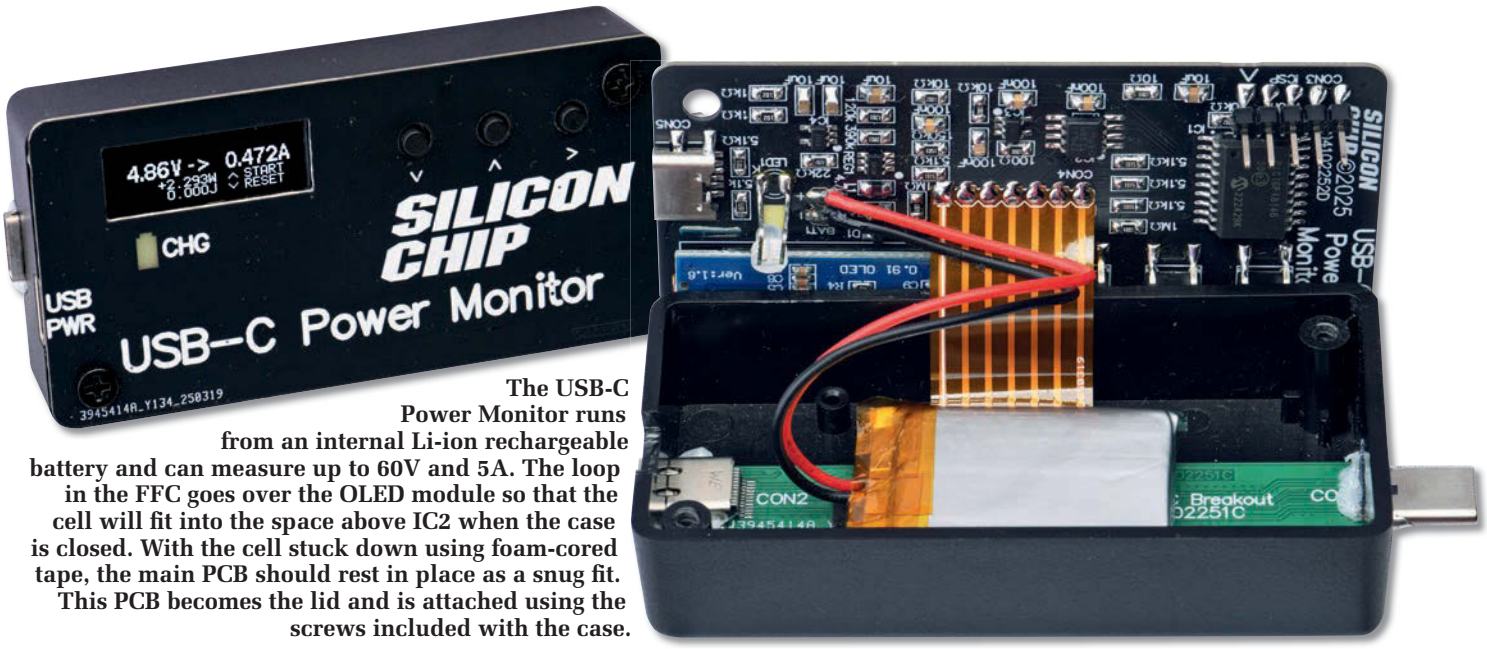
Once you are satisfied with the positions, solder the remaining leads. You can tidy up with a cotton-tipped swab or similar. Dip it in some solvent and wipe up any excess flux.

Programming IC1

Chips purchased from the Silicon Chip Online Shop (including those in the kit) will be supplied in a programmed state, so will not need programming. You can skip to the next section if you have such a chip.

To program IC1, you can fit a header to CON3. For our prototype, we just used a standard through-hole header strip and soldered it as though it were

USB-C Power Monitor, part two



The USB-C Power Monitor runs from an internal Li-ion rechargeable battery and can measure up to 60V and 5A. The loop in the FFC goes over the OLED module so that the cell will fit into the space above IC2 when the case is closed. With the cell stuck down using foam-cored tape, the main PCB should rest in place as a snug fit. This PCB becomes the lid and is attached using the screws included with the case.

a surface-mounting part. It can stay in place since it will not foul the enclosure if installed squarely. You can see it in the photo of the completed assembly above.

You can apply power to the PCB through CON5 (the USB-C power-only socket on the main PCB), if this is needed. Use a Snap, PICkit 4, PICkit 5 or PICkit Basic and choose the PIC16F18146 from the MPLAB IPE. Open the 0410225A.HEX file, use it to program the chip and confirm that the verification is successful.

You should also see the OLED illuminate and display the main screen; the readings will be nonsensical, since there is no connection to the second PCB. Pressing S3 (>) should cause the screens to cycle. If this is the case, then all is well. If not, double-check your soldering before continuing.

Enclosure

It's a good idea to prepare the case next, as that will allow the two PCBs to be checked before they are joined together. Note that the cutting here matches our prototype as built, with the USB-C socket (CON2) on the left and the plug (CON1) on the right. In other words, the CON5 USB-C power-only socket is on the same end as CON2.

Because the assembled PCB with CON1 & CON2 is longer than the case, the final assembly step will involve passing CON2 fully through its hole (and out the side of the case) so the CON1 end of the PCB can be dropped into place. The PCB is then slid back so that the two connectors are in their final locations.

Because of the extra room needed for these manoeuvres, the holes will be slightly oversized. As such, the lower PCB will need to be secured with glue (but not just yet!). This will also provide reinforcement against wear and tear on the connectors.

Fig.7 shows the cutting diagram, but you can also use the two PCBs to mark out the cuts as you go. For example, you can rest the main PCB on top of the case and mark the sides of CON5, rather than trying to measure out the dimensions with a ruler. The bottom of the holes for CON1 and CON2 should be level with the floor of the case, which is 2mm thick.

Since the CON5 cutout is a notch at the top edge of the enclosure, you could use a sharp hobby knife or fine hacksaw to make the vertical cuts. Make a score mark at the base of the tab and carefully flex it with pliers until it snaps off. Then tidy the edges and make sure that the main PCB can sit flat on the top of the case.

The other holes should be started with a drill at their outer edges. Make further drill holes along the length and use a file or knife to join those holes. Then enlarge the holes until the smaller PCB can be inserted and check that the lower PCB can slide freely and can be slotted into place from above.

Connecting the PCBs

The two PCBs are ideally connected by a flat flexible cable (FFC), which is effectively a flexible PCB with the code 04102253. An alternative is to simply use light-gauge insulated wiring or ribbon cable. The FFC is 4cm long, so wire, if used, should be the same length. This will neatly loop to one side of the case without bunching up. The loop provides space for the lithium cell.

Pay careful attention to the orientation of the two boards in our photos. You can see that CON2 is at the same end as CON5 to match our cutting diagram and prototype.

Start by soldering the FFC to the main PCB. We have aligned the striped conductor with the square pads on the PCB; this is pin 1. The FFC does not need to sit flat, but can pass at an angle just enough to clear the OLED module.

Just like an SMT part, you can tack one lead and confirm that the FFC is square and aligned to the main PCB. You shouldn't need to use flux since you will need to use a generous amount of solder. Tack the lead at the other end

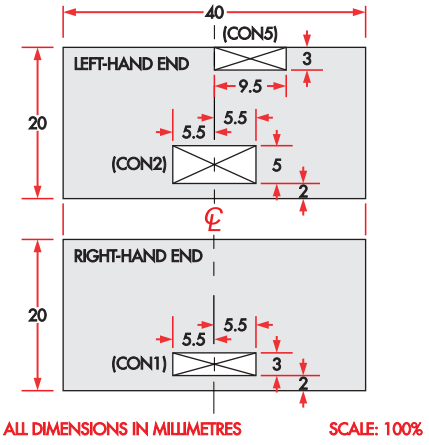


Fig.7: the recommended cut-outs. You will need to swap the CON1 & CON2 cut-outs if you plan to fit the smaller board in the opposite orientation than we are recommending.

and if everything looks correct, solder the remaining leads to their pads.

Follow the same process to connect it to the second PCB. The FFC connects to the side opposite the resistors, so it can sit flat against the PCB. You can see the arrangement in the photo on page 71.

Once this is done, take care with the assembly. The FFC is reasonably robust, but will not stand up to repeated flexing. It could tear if subjected to excessive force, or be kinked if it is bent too hard.

Completion

Now that the PCBs are connected, you could power up the unit (at CON5) and see that it is showing reasonable readings, close to 0V and 0A. Without a battery connected, our LED flashed red, then green and then switched off; if yours flashes green then red, the LED may be reversed, and it is best to correct that now.

You can jump ahead to the setup and usage section if you'd like to run some further checks before gluing the PCBs down and closing everything up.

Since the next step involves soldering the battery to the PCB, you should disconnect power. Take great care whilst working with the battery, since the lithium-ion cell will not take kindly to being short-circuited. Everything will be live (at up to 5V) from now on.

Carefully prepare the leads for soldering. Our battery had a connector that we needed to cut off. Only cut one lead at a time to avoid shorting them with the cutters. Use tape to cover the ends so that only one is exposed at a time. Solder the leads to the terminals marked BAT1 on the PCB, observing the polarity seen in the photos and on the silkscreen.

The Monitor should switch on. You can place it in a low-power sleep mode by pressing and holding S3 (>) until Screen 2 is seen. Then press S1 (down) to enter sleep; the text SLEEPING will appear before the screen blanks. Apply glue (neutral cure silicone) to the battery terminals on the PCB and cover any bare metal. This will add some extra strain relief and also insulate the bare ends of the wires.

Now you can slot CON1 and CON2 into the case. Ideally, CON2 should protrude slightly from its end, with the lower PCB resting against that end wall of the enclosure. Add glue to secure the PCB in place and take care not to allow any to seep inside the connectors, especially the holes on the top of CON2.

Screens 1-4: there are four main operating screens and nine configuration screens. These operating screens are described in detail in the text.

Screen 5: the brightness of the OLED screen can be adjusted here; the default value of 130 is near the midpoint of the adjustment range. Higher values will flatten the battery faster.

Screen 6: the displayed energy units on the main page can be set to either Joules (J) or Watt-hours (Wh). This can be changed at any time without affecting readings.

Screen 7: the TRIM factor on this screen sets the multiplier for voltage readings. Use a multimeter to compare the measured value against the displayed value.

Screen 8: the current ranges are trimmed in similar fashion to Screen 7. For the high current range, apply and measure a load of at least 1A to ensure accuracy across the range.

For now, apply just enough glue to make sure that the PCBs are mechanically secure. If needed, you can tidy up the external appearance by filling in the gaps in the case around CON1 and CON2 later. Now you should wait until the glue has fully cured to ensure that nothing breaks loose during the final stage.

While waiting for the glue to cure, you can charge the battery via the CON5 USB-C socket. The LED should light up red and then change to green when charging is complete.

Closing it up

Use foam-cored double-sided tape to secure the battery to the inside of the case. It should sit against the lower wall, near the middle of the case. Resting it against the internal boss should ensure that it is clear of CON3 if fitted. LED1 is the other component that might conflict, but that should not be a problem if you use the same size cell we did.

The main PCB is now placed on top of the case. Check that there aren't any internal collisions with the battery. If all is well, secure the lid with the two screws included with the case hardware. They will sit slightly above the surface of the PCB.

Don't screw them down too firmly; the thin PCB is flexible and will be somewhat susceptible to cracking if stressed.

Setup and usage

When the screen is blank, pressing any of the buttons should end the sleep mode. The unit returns to Screen 1 when this happens. A brief press of S3 (>) will cycle between the operating screens (Screens 1-4). A long press of S3 will enter the settings and configuration screens; there are nine of these, shown in Screens 5-13. You can exit the settings screens by another long press on S3.

The main screen (Screen 1) shows the measured voltage, current and power as well as accumulated energy. Below the current is a timer that can run up to 99 hours. On this screen, the up and down buttons control the timer and energy counter.

The state shown here has the timer stopped; the time display will alternate with ∧ START. Pressing ∧ will start the timer and the energy counter will integrate the power value. You can always calculate an average power by dividing the energy by the time.

Pressing ∨ will pause the timer (and energy counter) if it is running. Pressing it while paused will reset both values. The current display will show units of mA (to two decimal places) if the low current range is being used. The display will be in A (to three decimal places) when the higher range is in use.

The arrow on the first line shows the direction of current flow (source to sink). Right to left corresponds to a positive value of power and energy. Since the power will always be the same sign as the current, this should be unambiguous.

There is also a timeout that is only active on this screen. It is reset any time the V_{BUS} voltage is above 1V, if the timer is running or any time a button is pressed. If the timer is counting down, it is displayed in small text (along with the low V_{BUS} voltage) in the top left corner. The timer can be deactivated (as is the default), and we'll discuss this in the configuration section.

Sleep and battery

The next screen allows the battery voltage to be checked by pressing the ∧ button. This actually measures the voltage supplied to the micro and adds an adjustment for the diode, REG1 and 10Ω resistor. So it will only be accurate when there is nothing powering CON5.

This is on its own screen because it requires the boost regulator to be shut down and should not be done while the timer is running. It won't cause any damage, but the readings will be inaccurate since the 4.096V reference will not be at specification. The reading should be treated like a typical Li-ion battery voltage; 4.2V is close to fully charged and 3.6V or lower is flat.

Pressing the ∨ button on this screen will put the Monitor into low-power sleep mode. All timers and peripherals are shut down, as are REG1, MOD1, IC2 and IC3. If the timer from the main screen was running, it will be paused. The screen will show a SLEEPING message and then shut down. Pressing any of the buttons will wake it up.

Any time the Monitor is not being used, it should be put into sleep mode to avoid flattening the battery. The timeout on the main screen has the same effect and will show the same SLEEPING message.

CC states

The CC (configuration channel) lines are one of the new features that were introduced with USB-C. As we've noted in other articles, they have tripped up many engineers. So we thought that this screen (Screen 3) might help to shed some light on this feature. The V_{BUS} voltage is also displayed at lower right.

This screen depends on the connected devices complying with the standards, so if you see nonsensical readings, maybe there is a problem with whatever is connected to CON1 and CON2. It's also possible that the 220Ω resistors in the Monitor are interfering with its operation, although they generally shouldn't.

The second line shows which of the two (CC1 and CC2) lines is used for CC signalling on the connected USB-C source; this corresponds to either the upper (A5) or lower (B5) connections on CON1 or CON2. This can help with troubleshooting cable orientation. A sink device must be connected to provide the pulldown on the CC lines before the source current can be read.

If the text ∧ START is shown on the last line, the Monitor can provide that sink by pulling its internal 5.1kΩ resistors low. The ∧ button must be held down to apply the internal 5.1kΩ load and no other sink should be connected; this is the reason for the SOURCE ONLY warning.

When a sink is provided, the second and third lines provide information about the source capabilities and status. You should see either → or ← pointing from source to sink, and some text describing the status. The direction is derived from the current through the 220Ω resistors.

The status is derived from the voltage on the active CC line and the V_{BUS} voltage. For example, a V_{BUS} voltage over 5.5V is interpreted as a USB-PD voltage being negotiated; this is displayed with the text USB-PD.

A LEGACY source is one that applies V_{BUS} without a sink being connected. This implies a USB-A to USB-C cable or adaptor has been used on the source side. You might also see SOURCE LOW if the Monitor determines the source should be supplying 5V but is not.

We have found that some devices don't respond instantly to changes in the configuration channel. Some of the timeouts in the USB-C specification allow over a second for some responses to occur, so this is to be expected.

Screen 9: the low-current range works up to about 25mA, so a 220Ω resistor across 5V will provide an appropriate load for trimming on this page.

Screen 10: the current offsets can be automatically trimmed by the page shown in Screen 4, but the value is shown here for completeness.

Screen 11: you can also manually trim the current offsets by adjusting the parameter until the displayed current reads zero, as shown here.

Screen 12: by default, the display timeout is disabled, but it can be switched on by adjusting the value upwards. The timeout only applies on the main screen if the Monitor is idle.

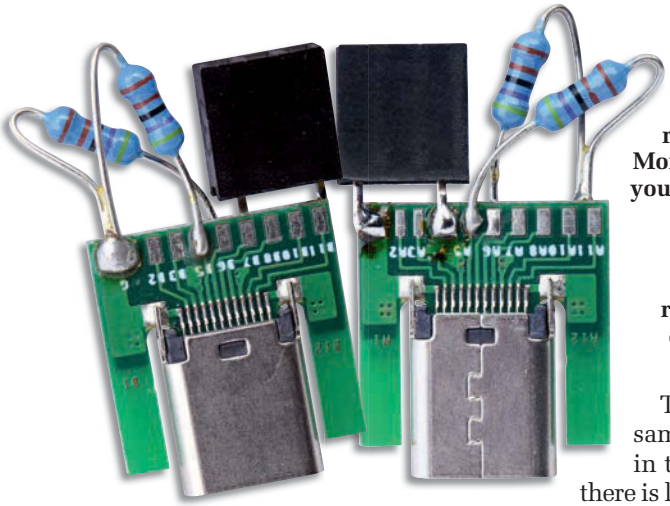
Screen 13: the configuration is held in RAM, which will be lost if the battery runs flat. So we recommend you perform a SAVE after doing the initial calibration.

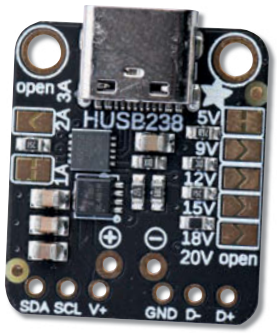
Offset trim

Screen 4 shows a page used to trim the offset in the current-measuring channels. The offset changes with V_{BUS} voltage, so it is best to have the expected voltage present when doing this. The default (zero) trim values are fairly accurate at 5V, since this is near the supply voltage of IC2 and IC3 involved in current measurement.

When the ∨ button is pressed, the Monitor takes an average over 256 readings of the high and low current ranges. It then applies this as the offset to the raw ADC value, as shown at the bottom of the screen.

As the text explains, the current





Left: this Adafruit 4396 USB-C socket breakout board is fitted with two 5.1kΩ resistors and a header. It will be handy during Monitor calibration and could also be useful if you need a 5V supply with a current readout.

Right: a typical USB-PD trigger board has a USB-C socket, a USB-PD interface IC and an output connector. This example sets the requested PD voltage by solder jumpers; some can be controlled digitally, with an I²C serial interface or similar.

should be zero for this to work correctly. If there is an idle or quiescent current that you wish to cancel out, this should be applied, and it can be trimmed out, too. An example of this is the load due to the V_{BUS} sensing divider.

Configuration

Screens 5-13 are configuration screens. Most of these screens are fairly straightforward, and there are brief descriptions of each in the captions. The three TRIM screens adjust the multiplier used in calculating the V_{BUS} voltage and high and low current ranges. The offset trim described above should be done before completing this step using the same V_{BUS} voltage.

You'll need a multimeter or similar so that you can read a value to trim against. The parameter shown in the second line should be adjusted until the measured value (volts or amps) matches the displayed value. For the current ranges, you might see INVALID displayed if the Monitor thinks the analog voltage is near its limit or saturating; this is most likely on the low current range.

The OFFSET pages are the same parameters as described in the Offset trim section, and there is little need to manually adjust these. They are simply provided for completeness.

The Monitor will use the live settings at all times, although Screen 13 shows a page to save the settings to EEPROM. Since the Monitor has the battery permanently connected, there is little chance of the Monitor forgetting its settings in RAM. But if the battery were to run flat, it would do so.

So we recommend you save the settings to EEPROM using the ∧ button once the Monitor is set up. If you ever have a problem with the settings being corrupted, the ∨ button will restore to active settings from the defaults in flash memory. You can then save these to EEPROM with the ∧ button to complete the RESTORE.

Accessories

During testing, we used various cables, adaptors and breakout boards to test and probe the operation of the Monitor. You'll need a standard USB-C plug-plug cable to use the Monitor, just as you would need such a cable to operate the device you are testing.

A small breakout board like Adafruit's 4396 USB Type C Socket breakout (shown in the left-hand photo) could be handy for calibration. Any similar breakout board that exposes V_{BUS}, GND and the CC lines of a USB-C socket should also work.

We wired up the two CC lines to allow the breakout to behave as a sink; there is a 5.1kΩ resistor from pin A5 to ground and another 5.1kΩ resistor from B5 to ground. A three-way header socket with the middle pin removed has the right pitch to connect to ground and V_{BUS}. The photos at upper left show this gadget from both sides.

This can be used for calibration, as we mentioned earlier, or to turn the Monitor into a current and voltage display for a simple 5V power supply. If you need access to higher voltages, a USB-PD trigger board (as seen in the photo above) might be an alternative.

Conclusion

The USB-C Power Monitor is a necessarily more complex design than its predecessor from 2012. It allows monitoring of the higher currents and voltages that USB-C allows. It can also provide information about newer features specific to USB-C.

After it is set up, operation is straightforward. Typically, you would connect your device to its host or power supply using a standard USB-C plug-plug cable. The Monitor is fitted inline with the device to be checked. USB-C's reversible plug and socket mean that you have some flexibility in how it is connected.

Once you have adjusted the trim offsets to your liking, you can monitor the current, voltage & power. Starting the timer will allow you to check total energy consumption over a period and thus also average power consumption.

The CC Connection State page (Screen 3) allows you to check the behaviour of USB-C's configuration channel. Make sure to put the unit into low-power sleep when you are finished so that the battery does not run flat. PE

Features & Specifications

- Main screen reports current, voltage, power, energy (in J or Wh) & time
- Configuration channel (CC) status screen
- All 24 USB data lines pass through
- Self-contained with 400mAh rechargeable lithium battery
- Internal battery means no extra load on the USB circuit under test
- Compact case is only 80 × 40mm
- Automatic offset trimming
- Voltage measurements: up to 60V with 10mV resolution
- Current measurements: up to ±5A with 1mA resolution; 10µA resolution below ~25mA
- Power: up to 300W with 1mW resolution (limited by V and I)
- Energy: up to 999999J (1mJ resolution) or up to 999Wh (10µWh resolution)
- Battery consumption: <20mA, giving 20 hours of usage per charge
- Sleep mode: <10µA drawn from battery, less than typical self-discharge

Techno Talk, continued from p67...

Instead of simply projecting flat images into your eyes, Swave uses chip-based holography to sculpt the light itself. Around a quarter of a billion pixels per eye create genuine three-dimensional images that appear naturally integrated into the world around you.

Swave's microscopic 'nanopixels' are small enough to manipulate light waves directly, eliminating much of the bulky optics used in today's AR systems while opening the door to compact, low-power smart glasses.

Will this become the display technology of the future? I have no idea. But if history has taught us anything, it's that engineers possess an uncanny knack for inventing wonderfully ingenious ways of getting information in front of our eyeballs. From gears and springs to glowing phosphors and holographic light fields, it has been an amazing journey so far. I suspect the best is yet to come!

I can see something like this in our future (Source: Swave)

Cool as a cucumber

My wife, Gigi the Gorgeous, has a love-hate relationship with technology. She loves to use it, but she hates it when it doesn't work. This is a tad ironic because high-tech devices tend to work perfectly well… right up until Gigi uses them. This explains why I don't let her anywhere near any of the computers I use for work.

One thing Gigi hates is when her smartphone gets almost too hot to hold. Why does this happen? My own phone, which is a much older model, never even gets warm. The answer is that I use my phone only sparingly for short, focused conversations.

By comparison, Gigi can easily stretch a topic-less call with her mum (or with an AI chatbot, for that matter) over several hours, which requires her to connect a battery pack to keep the phone charged and the call alive.

As a result, the phone gets unpleasantly warm, prompting Gigi to share her displeasure with me. Apparently, all engineers are jointly responsible for the laws of physics. We're sorry; we'll try not to do it again.

Happily, help is on the way. I was just chatting with the guys and gals at Boréas Technologies (**www.boreas.ca**). Their latest technology essentially brings the active liquid-cooling techniques inspired by modern AI data centres into devices small enough to fit in your pocket.

Instead of relying on passive cooling techniques like heat spreaders and vapour chambers, Boréas combines a tiny piezoelectric micropump with an ultra-thin microchannel cooling plate. A liquid coolant is actively circulated through microscopic channels close to the processor, transporting heat away from local hot spots and distributing it more evenly throughout the device.

The result isn't necessarily a colder phone overall; it's one without the annoying hot spots that make you feel as though your ear is about to burst into flames ("I'm sorry, can you repeat that… aaarrgghhh!").

Virtual acoustic worlds

I'm afraid my mind is roaming wild and free, as is its wont. Talking about people talking on smartphones reminded me of a conversation I recently had with the folks at the Icelandic startup Treble Technologies (**www.treble.tech**).

Since we rely so much on our visual perception, it's easy to forget that we don't experience the world using our eyes alone. Our ears constantly tell us where things are, how far away they are, and what sort of environment we're in.

When talking on the phone, for example, we instinctively know whether the person at the other end of the call is speaking to us from a cosy living room, a tiled bathroom, a cathedral, or the middle of a forest. That's because every space leaves its own acoustic fingerprint.

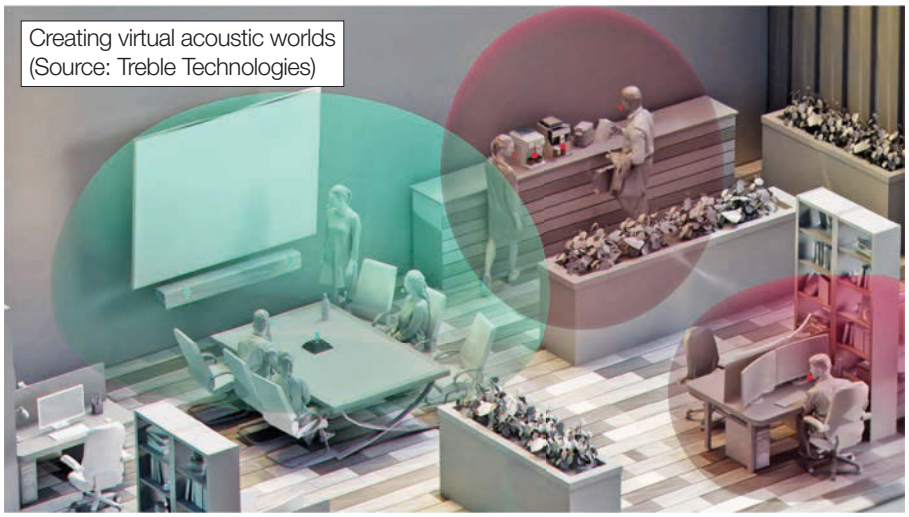
Treble has developed a technology that can model those acoustic fingerprints with remarkable accuracy. Instead of simply making sounds louder or quieter, their software predicts how sound waves will reflect from walls, ceilings, furniture, windows and almost anything else you can imagine. The result is a virtual acoustic environment that's astonishingly lifelike.

This may sound like something intended solely for architects and audio engineers, but I think its real potential lies elsewhere. Imagine combining future AI glasses with this technology.

Instead of merely seeing a virtual arrow pointing towards your destination, you might hear a virtual guide walking beside you.

Rather than listening to a flat, lifeless phone call, the other person could be made to sound as though they were actually sitting across the table from you. As our computers become ever better at seeing the world around us, it's only fitting that they should also become better at hearing it.

Looking back over the past century or so, it's remarkable how technology has evolved. First, we taught our machines to display information, and now we're teaching them to see, hear, and understand the world around them. I have absolutely no idea where all this is leading, but it's going to be a lot of fun finding out! PE


Creating virtual acoustic worlds (Source: Treble Technologies)

Practical Electronics PCB SERVICE

PROJECT	CODE	PRICE
SEPTEMBER 2026		
Home Automation Satellite PCB	15104251	£2.99
Mic the Mouse PCB set	08105251-2	£3.99
AUGUST 2026		
RP2350B Computer (assembled)	07204251A	£54.99
USB-C Power Monitor PCB set (including flex cable)	04102251-3	£6.99
Audio Out Phase-Linear Crossover PCB	AO-AUG-26	£4.99
VFD Chronometer digit board	CBAUG26-1	£2.99
VFD Chronometer spacer board	CBAUG26-2	£2.99
JULY 2026		
RP2350B Development Board (assembled)	07107251A	£24.99
SmartProbe PCB	P9054-04	£4.99
SWD Adaptor board for the SmartProbe	P9045-A	£2.99
Audio Out New All-Pass Filter PCB	AOJUL26-1	£4.99
Audio Out Alps Polarity Switch PCB	AOJUL26-2	£2.99
Audio Out E-switch Polarity Switch PCB	AOJUL26-3	£2.99
JUNE 2026		
RGB LED 'Analog' Clock PCB (black)	19101251	£7.99
USB Power Adaptor PCB (set of two)	18101251	£2.99
433MHz Receiver Module PCB	15103252	£2.99
Audio Out All-Pass Filter PCB set (three PCBs)	AO-JUN-26	£6.99
MAY 2026		
SSB Shortwave Receiver three PCB set	CSE250202-4	£15
Tool Safety Timer PCB	10104251	£3.99
433MHz Transmitter Module PCB	15103251	£2.99
APRIL 2026		
Versatile Battery Checker PCB	11104251	£3.99
Versatile Battery Checker front panel (black, 0.8mm)	11104252	£4.99
Power Rail Probe PCB	P9058-1-C	£3.99
Pico 2 Audio Analyser PCB/panel (black)	04107231	£3.99
MARCH 2026		
USB-Programmable Frequency Divider PCB (blue)	04108241	£3.99
Rotating Light PCB (black)	09101251	£2.99
433MHz Transmitter PCB	15103251	£2.99
Arduino Uno Game Cartridge PCB (Cool Beans)	CBmar26uno	£3.99
Arduino Nano Game Cartridge PCB (Cool Beans)	CBmar26nano	£2.99
Silent Stomp Switch PCB pair (Audio Out)	AO-MAR26-*	£3.99
FEBRUARY 2026		
Power LCR Meter PCB	04103251	£7.99
Versatile Waveform Generator PCB	04104251	£5.99
JANUARY 2026		
High-Bandwidth Differential Probe	9015-D	£5.99
NFC-Programmable IR Remote Keyfob	15109231	£2.99
Pre-driver Overdrive Pedal main PCB	AO-JAN26-1	£4.99
Updated jack PCB for the Pre-driver	AO-JAN26-2	£3.99
DECEMBER 2025		
Variable Speed Drive (black)	11111241	£11.99
Digital Capacitance Meter	04111241	£5.99
Battery-Powered Model Train transmitter	09110241	£3.99
Battery-Powered Model Train through-hole receiver	09110242	£3.99
Battery-Powered Model Train SMD receiver	09110243	£3.99
Battery-Powered Model Train battery charger	09110244	£3.99
Transmanium Fuzz Box jack PCB	AO-DEC-25	£3.99
NOVEMBER 2025		
Capacitor Discharger	9047-01	£4.99
5MHz, 40A Current Probe	9049-01	£5.99
Transmanium Fuzz Box main PCB	AO-NOV-25	£4.99

PROJECT	CODE	PRICE
OCTOBER 2025		
Seaside Sound Simulator PCB (blue)	01111241	£7.99
Fender Jazz Bass pickguard	23109241	£5.99
J&D Luthiers T-Style pickguard	23109242	£5.99
Music Man Stingray pickguard	23109243	£5.99
Fender Telecaster pickguard	23109244	£4.99
3D Printer Filament Dryer control PCB	28110241	£4.99
SEPTEMBER 2025		
Automatic LQ Meter PCB	CSE240203A	£5.99
Automatic LQ Meter front panel (black)	CSE240204A	£5.99
Compact Hi-Fi Headphone Amplifier PCB (blue)	01103241	£4.99
Discrete Ideal Bridge Rectifier TH or SMD PCB	18108241/2	£2.99
Dual Rail Supply Protector PCB	18109241	£3.99
Audio Out Cascode Amplifier PCB	CASCODEA	£4.99
Audio Out Cascode Amplifier Power Supply PCB	CASCODEP	£4.99
AUGUST 2025		
Micromite Explore-40 PCB	07106241	£2.99
Micromite Explore-40 Audio Breakout PCB	07101222	£2.99
8-Channel Learning IR Remote Receiver PCB	15108241	£5.99
FlexiDice PCB set	08107241-2	£4.99
JULY 2025		
Compact OLED Clock & Timer PCB	19101231	£4.99
180-230V DC Motor Speed Controller PCB	11104241	£9.99
Mains Power-Up Sequencer PCB (same as Feb '25)	10108231	£11.99
JUNE 2025		
Styloclone (case version)	23106241	£6.99
Styloclone (standalone version)	23106242	£7.99
Dual Mini LED Dice (TH LEDs, white solder mask)	08103241W	£2.99
Dual Mini LED Dice (TH LEDs, black solder mask)	08103241B	£2.99
Dual Mini LED Dice (SMD LEDs, white solder mask)	08103242W	£2.99
Dual Mini LED Dice (SMD LEDs, black solder mask)	08103242B	£2.99
Dual Hybrid Supply regulator board	18107211	£5.99
Dual Hybrid Supply front panel control board	18107212	£2.99
Dual Hybrid Supply CPU board	01106193	£3.99
Dual Hybrid Supply LCD adaptor	01106196	£2.99
MAY 2025		
Skill Tester 9000	08101241	£9.99
USB-C Serial Adaptor (black solder mask)	24106241	£2.99
DC Supply Protector (SMD adjustable version)	08106241	£3.99
DC Supply Protector (TH adjustable version)	08106242	£3.99
DC Supply Protector (TH fixed version)	08106243	£3.99
APRIL 2025		
ESR Test Tweezers PCB set	04105241+	£5.99
Fan Speed Controller Mk2	10104241	£4.99
Calibrated Microphone (SMD version)	01108231	£3.99
Calibrated Microphone (TH version)	01108232	£3.99
MARCH 2025		
Pico Gamer	08104241	£9.99
Compact Frequency Divider	04112231	£4.99
FEBRUARY 2025		
Laser Communicator transmitter	16102241	£3.99
Laser Communicator receiver	16102242	£3.99
Model Railway Points Controller transmitter	09101241	£3.99
Model Railway Points Controller receiver	09101242	£2.99
Mains Power-Up Sequencer	10108231	£11.99

PCBs for most recent *PE/EPE* constructional projects are available. From the July 2013 issue onwards, PCBs with eight-digit codes have silk screen overlays and, where applicable, are double-sided, have plated-through holes, and solder mask. They are similar to photos in the project articles. Earlier PCBs are likely to be more basic and may not include silk screen overlay, be single-sided, lack plated-through holes and solder mask.

Always check price and availability in the latest issue or online. A large number of older boards are listed for ordering on our website.

In most cases we do not supply kits or components for our projects. For older projects it is important to check the availability of all components before purchasing PCBs.

Magazine back issues are available – see https://www.electronpublishing.com/product-category/pe-back-issues-print-and-digital/

Double-sided | plated-through holes | solder mask

PROJECT	CODE	PRICE
JANUARY 2025		
USB to PS/2 Converter for VGA PicoMite	07111231	£3.99
ps2x2pico USB to PS/2 Converter	07111232	£3.99
Compact Microphone Preamplifier (case version)	01110231	£5.99
Compact Microphone Preamplifier (integrated version)	01110232	£5.99
DECEMBER 2024		
Raspberry Pi Clock Radio PCB pair	19101241/2	£12.99
VGA PicoMite (blue)	07107221	£5.99
Secure Remote Switch receiver	10109231	£4.99
Secure Remote Switch transmitter (module version)	10109232	£2.99
Secure Remote Switch transmitter (discrete version)	10109233	£2.99
NOVEMBER 2024		
Ideal Diode Bridge Rectifiers (28mm square)	18101241	£3.99
Ideal Diode Bridge Rectifiers (22mm square)	18101242	£3.99
Ideal Diode Bridge Rectifiers (4-pin SIL)	18101243	£3.99
Ideal Diode Bridge Rectifiers (tiny SOT-23)	18101244	£3.99
Ideal Diode Bridge Rectifiers (standalone SMD)	18101245	£4.99
Ideal Diode Bridge Rectifiers (standalone TH; 2oz copper)	18101246	£6.99
Multi-channel Volume Control (volume PCB)	01111221	£4.99
Multi-channel Volume Control (control PCB)	01111222	£4.99
Multi-channel Volume Control (OLED PCB, black)	01111223	£4.99
Coin Cell Emulator (black)	18101231	£3.99
OCTOBER 2024		
TQFP-32 Programming Adaptor	24108231	£4.99
TQFP-44 Programming Adaptor	24108232	£4.99
TQFP-48 Programming Adaptor	24108233	£4.99
TQFP-64 Programming Adaptor	24108234	£4.99
2m VHF FM Test Signal Generator	06107231	£4.99
SEPTEMBER 2024		
PIC Programming Adaptor	24105231	£4.99
Salad Bowl Speaker crossover	01109231	£7.99
AUGUST 2024		
Arduino ESR Meter combined L/C-ESR	04106182	£9.99
Arduino ESR Meter separate ESR	04106183	£7.99
Watering System Controller	15110231	£11.99
JULY 2024		
Dynamic NFC Tag (22 × 31mm)	06101231	£5.95
Dynamic NFC Tag (86 × 54mm)	06101232	£7.95
Reciprocal Frequency Counter	CSE230101C	£8.95
Reciprocal Frequency Counter front panel (black mask)	CSE230102	£7.95
Pi Pico-based Thermal Camera	04105231	£6.95
JUNE 2024		
Loudspeaker Test Jig	04106231	£12.95
MAY 2024		
GPS-Disciplined Oscillator	04103231	£12.95
Dual RF Amplifier	CSE220602A	£5.95
Songbird	08103231	£7.95
APRIL 2024		
Basic RF Signal Generator	CSE221001	£8.99
Basic RF Signal Generator front panel (black solder mask)	CSE220902B	£9.99
Silicon Chirp Cricket	08101231	£7.99
Test Bench 'Swiss Army Knife'	04110221	£9.99
MARCH 2024		
Digital Volume Control Potentiometer SMD version	01101231	£8.95
Digital Volume Control Potentiometer through-hole version	01101232	£9.95
Advanced SMD Test Tweezers body	04106221	£6.95
Advanced SMD Test Tweezers legs (pair)	04106212	£7.95

PROJECT	CODE	PRICE
FEBRUARY 2024		
Active Mains Soft Starter	10110221	£10.95
JANUARY 2024		
Q Meter	CSE220701	£8.95
Q Meter front panel (black solder mask)	CSE220704	£7.95
Raspberry Pi Pico W BackPack	07101221	£8.95
DECEMBER 2023		
Digital Boost Regulator	24110224	£8.95
Dual-Channel Power Supply for Breadboards	04112221	£11.95
Display Adaptor for the Breadboard PSU	04112222	£9.95
NOVEMBER 2023		
LC Meter Mk3	CSE220503C	£9.95
DC Supply Filter for vehicles	08108221	£8.95
Discrete Audio Op Amp PCB	AO1-JUL23	£9.95
Discrete Audio Op Amp PCB + essential components	AO2-JUL23	£17.95
OCTOBER 2023		
Buck/Boost Charger Adaptor	14108221	£9.95
PIC Breakout Board for SOIC parts	24110225	£5.95
PIC Breakout Board for DIP parts	24110222	£5.95
AVR64DD32 Breakout board	24110223	£5.95
Automatic Train Controller	09109221	£6.95
Chuff Sound module	09109222	£6.95
SEPTEMBER 2023		
Mini LED Driver	16106221	£7.95
New GPS-Synchronised Clock	19109221	£9.95
AUGUST 2023		
110dB RF Attenuator	CSE211003	£9.95
Universal Battery Charge Controller (2023 update)	14107192	£9.95
Wide-Range OhmMeter	04109221	£12.95
JULY 2023		
Multimeter Checker/Calibrator	04107221	£10.95
JUNE 2023		
Arduino Programmable Load	04105221	£8.95
Buck-Boost LED Driver	16103221	£8.95
MAY 2023		
Precision AM-FM DDS Signal Generator	CSE211002	£14.95
Improved SMD Test Tweezers programmed PIC	0410621PIC	£12.95
AO 2x-dual-to-quad through-hole (pack of five)	AO1-MAY23	£6.95
AO 2x-dual-to-quad SMD (pack of five)	AO2-MAY23	£6.95
AO 2x-single-to-dual through-hole (pack of five)	AO3-MAY23	£6.95
AO 2x-single-to-dual SMD (pack of five)	AO4-MAY23	£6.95
APRIL 2023		
500W Amplifier Module	see p.22, April 2023	
Clipping Indicator (per channel)	01112211	£7.95
CD Welder Power Supply (one needed)	29103221	£9.95
CD Welder Controller (one needed)	29103222	£9.95
CD Welder Energy Storage module (several needed)	29103223	£7.95
AO Universal Dual Op Amp Board	AO1-APR23	£9.95
AO Stereo RIAA precision passives kit for dual op amp	AO2-APR23	£8.95
MARCH 2023		
Pico BackPack	07101221	£9.95
Semaphore Signal (controller)	09103221	£7.95
Semaphore Signal (blade)	09103222	£5.95

All prices include VAT and UK p&p. Add £4 per *project* for post to Europe; £5 per *project* outside Europe.

Orders and payment should be sent to:

Practical Electronics, Electron Publishing Ltd
113 Lynwood Drive, Merley, Wimborne, Dorset BH21 1UU
Tel 07918 614662 Email: pe@pemag.au
Online Shop: www.electronpublishing.com

Sorry, cheques cannot be accepted as we are based in Australia and our UK bank account does not allow cheque deposits.

NOTE: Most boards are in stock and sent within seven days of order receipt. Please allow up to 28 days delivery if we need to restock.

Double-sided | plated-through holes | solder mask

PROJECT	CODE	PRICE
FEBRUARY 2023		
Tesla Coil driver board	26102221	£9.95
Tesla Coil potentiometer board	26102222	£5.95
Cooling Fan Controller & Loudspeaker Protector	01102221	£8.95
Remote Gate Controller	11009121	£12.95
JANUARY 2023		
Classic LED Metronome – 8-LED	23111211	£7.95
Classic LED Metronome – 10-LED	23111212	£8.95
Multi-Channel Speaker Protector – 6 channel	01101221	£9.95
Multi-Channel Speaker Protector – 4 channel	01101222	£7.95
Remote Control Range Extender – IR-to-UHF	15109212	£5.95
Remote Control Range Extender – UHF-to-IR	15109211	£7.95
AO Universal Single Op Amp Board	AO1-JAN23	£7.95
DECEMBER 2022		
Hummingbird Amplifier	01111211	£9.95
SMD Trainer PCB	29106211A	£8.95
NOVEMBER 2022		
USB Cable tester – main PCB	04108211	£12.95
USB Cable tester – front panel	04108212	£5.95
USB Cable tester – optional panel	SC5970	£5.95
Model Railway Carriage Lights – PCB	09109211	£6.95
AO trnsfmr PCB – standard VTX-A range	VTX-101-007	£6.95
AO trnsfmr PCB – dual-outline VTX102-3001/101-3002	VTX-Dual	£6.95
OCTOBER 2022		
SMD Test Tweezers – PCB and pair of tweezer arms	04106211-2	£11.95
SMD Test Tweezers – programmed PIC12F1572-I/SN	0410621A	£7.95
Tele-com	12110211	£12.95
SEPTEMBER 2022		
Touchscreen Digital Preamp – main board	01103191	£12.95
Touchscreen Digital Preamp – adaptor board pair	01103192	£5.95
20A DC Motor Speed Controller	11006211	£9.95
AUGUST 2022		
Multi-purpose Battery Manager – I/O Expander	11104212	£5.95
Multi-purpose Battery Manager – Switch Module	11104211	£8.95
Simple MIDI Music Keyboard (for 8 switches)	23101213	£6.95
Nano Pong	08105212	£7.95
JULY 2022		
Silicon Labs AM/FM/SW Radio	CSE210301C	£10.95
Level Crossing Controller	09108211	£6.95
JUNE 2022		
Full-wave Motor Speed Controller	1010221	£8.95
PIC Programming Helper for 8-pin PICs only	24106211	£7.95
PIC Programming Helper for 8, 14 or 20-pin PICs	24106212	£10.95
Advanced GPS Computer	05102211	£9.95
MAY 2022		
Bus board PCB for Analogue Vocoder	AO1-MAY22	£10.95
Complete set of 14 PCBs for Analogue Vocoder	AO2-MAY22	£97.95
Programmed EEPROM for Digital FX Unit	FX1-MAY22	£10.95
Programmed PIC for Digital FX Unit w/ potentiometer	FX2-MAY22	£8.95
APRIL 2022		
64-key MIDI Matrix shield	23101211	£8.95
64-key MIDI Matrix switch board	23101212	£11.95
High-current Battery Balancer	14102211	£10.95
Digital FX Unit – using potentiometer	01102211	£9.95
Digital FX Unit – using BCD switch	01102212	£9.95
Universal Audio PSU	AO1-APR22	£11.95

PROJECT	CODE	PRICE
MARCH 2022		
Mini Isolated Serial Link	24102211	£5.95
Busy Loo Indicator	16112201	£5.95
Analogue Vocoder – Band-pass filter board	AO1-MAR22	£9.95
Analogue Vocoder – HP/LP filter board	AO2-MAR22	£9.95
FEBRUARY 2022		
Arduino-based Power Supply	18106201	£9.95
Battery Monitor Logger	11106201	£10.95
Electronic Wind Chimes	23011201	£10.95
Analogue Vocoder – Driver Amplifier	AO-FEB22	£8.95
JANUARY 2022		
Vintage battery Radio Li-ion Power Supply	11111201	£9.95
MiniHeart: A Miniature Heartbeat Simulator	01109201	£8.95
DECEMBER 2021		
AM/FM/SW Digital Receiver	CSE200902A	£13.95
Balanced Input and Attenuator for USB CODEC	01106202	£11.95
NOVEMBER 2021		
Dual Battery Lifesaver	11111202	£6.95
OCTOBER 2021		
Mini Wi-Fi LCD BackPack	24106201	£8.95

For the many earlier PCBs that we stock please see the PE website: www.electronpublishing.com

PE/EPE PCB SERVICE

Order Code Project Quantity Price

. . . . . .

. . . . . .

. . . . . .

. . . . . .

. . . . . .

. . . . . .

. . . . . .

Name . . . . . .

Address . . . . . .

. . . . . .

Tel . . . . . .

Email . . . . . .

Card No . . . . . .

Valid From . . . . . Expiry Date . . . . . .

Card Security No . . . . . MasterCard VISA

You can also order PCBs by phone, email or via the shop on our website: www.electronpublishing.com

No need to cut your issue – a copy of this form is just as good!

Kits and pre-programmed microcontrollers are available from the Silicon Chip shop in Australia for many of the projects published in *Practical Electronics*. Kits relevant to article in this issue:

USB-C Power Monitor short-form kit (includes everything but case, battery & glue)	https://siliconchip.au/Shop/20/7489	AU$72 (~£38) + P&P
Mic the Mouse kit (nicludes everything but the cell)	https://siliconchip.au/Shop/20/7508	AU$45 (~£23) + P&P

To see a complete list of the kits available, visit: https://siliconchip.au/Shop/20
Pre-programmed microcontrollers are listed here: https://siliconchip.au/Shop/9

VAT is charged to UK customers at check-out from the Silicon Chip shop and currently, orders under £85 should not attract any extra duties. Please allow 1-2 weeks for delivery (longer in some cases).

CLASSIFIED ADVERTISING

Practical Electronics

If you want your advertisements to be seen by the largest readership at the most economical price then our classified page offers excellent value.

The rate for semi-display space is £10 per centimetre high, with a minimum height of 2·5cm. All semi-display advertisements have a width of 5.5cm. The prepaid rate for classified adverts is 40p per word (minimum 12 words).

For rates, further information on display and classified advertising or to book an ad, please email us at **pe@pemag.au**

Practical Electronics reaches more UK readers than any other UK monthly hobby electronics magazine. Our sales figures prove it.

We have been the leading monthly magazine in this market for more than thirty years.

W5SWL Electronics

Technical Products and Small Component Parts, RF and Audio Connectors, Hobby Electronics, Ham Radio to Microwave Products, General Hobby Supplies, Technical Books, Tools, and More. **Located in the United States.**

Visit **www.W5SWL.com**

BOWOOD ELECTRONICS LTD

Suppliers of Electronic Components

www.bowood-electronics.co.uk

Unit 10, Boythorpe Business Park, Dock Walk, Chesterfield, Derbyshire S40 2QR. Sales: 01246 200 222

Send large letter stamp for Catalogue

COAST ELECTRONICS

BREAKOUTS-COMPONENTS-CONTRACT DESIGN-3D PRINTER PARTS-MUSICAL-MICROCONTROLLERS

WWW.COASTELECTRONICS.CO.UK

MISCELLANEOUS

VALVES AND ALLIED COMPONENTS? For free stock list and/or advice, please contact me: geoffdavies337@gmail.com Telephone: 01788 574774

PIC DEVELOPMENT KITS, DTMF kits and modules, CTCSS Encoder and Decoder/Display kits. Visit www.cstech.co.uk

Electrical Industries Charity (EIC)
We help people working in the electrical, electronics and energy community as well as their family members and retirees.

We use workplace programmes that give the industry access to financial grants and a comprehensive range of free and confidential services.

www.electricalcharity.org

ADVERTISING INDEX

AOSHOP 33
ENIGMA COMPONENT SHOP. 16
HAMMOND MANUFACTURING. 25
MICROCHIP TECHNOLOGY . . . . . Outer back cover
PCBWAY. . . . . Inner front cover
PDF BACK ISSUE DOWNLOADS 45
PO LABS 57
TAG CONNECT 16
TEACH-IN BUNDLES & DVDS . . . . . 32 & 66

Advertising mailing address

Electron Publishing Australia
PO Box 194
Matraville NSW 2036
Australia

Email pe@pemag.au

Web www.electronpublishing.com

For editorial contact details, see page 2.

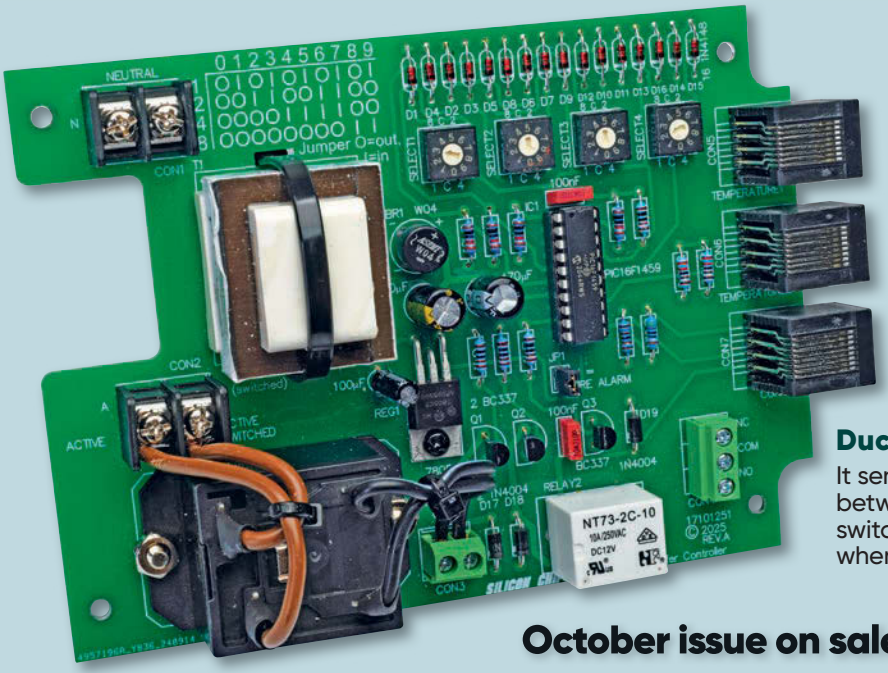
Coming Up – in the Oct/Nov issues

DIgital Preamplifier & Crossover

A comprehensive preamplifier with four stereo analog inputs, two digital inputs and USB audio in/out. It uses high-performance digital signal processing (DSP) for all functions.

Power Rail Probe

This inexpensive device makes it easier to monitoring power supply rails for ripple and noise with an oscilloscope.



Ducted Heat Transfer Controller

It senses the difference in temperature between rooms and automatically switches on a fan to transfer warm/cool air when required. It's highly configurable.

October issue on sale: 4th of September, 2026

Content may be subject to change

Practical Electronics

Subscriptions hotline

To subscribe or renew online, paying by credit/debit card or PayPal, please visit our shop at: www.electronpublishing.com

For those wishing to pay by direct debit, cheque or over the telephone, please call 'Select Publisher Services' on 01202 087631, or send a cheque (payable to: 'Select Publisher Services Ltd') with your details to:

Practical Electronics Subscriptions, PO Box 6337, Bournemouth BH1 9EH, United Kingdom

Remember, we print the date of the last issue of your current subscription in a box on the address sheet that comes with your copy.

Digital subscribers, please call 07918 614662 or visit: www.electronpublishing.com

Published on approximately the first Thursday of each month by Silicon Chip Publications, PO Box 194 Matraville NSW 2036 Australia. Printed in England by The Manson Group, Reynolds House, 8 Porters Wood, Valley Road Industrial Estate, St Albans AL3 6PZ. Distributed by Seymour, 86 Newman St, London W1T 3EX. Subscriptions UK: £33.99 (6mo); £59.99 (12mo.); £114.99 (2yrs). EUROPE: airmail service, £38.99 (6mo); £69.99 (12mo); £129.99 (2yrs). REST OF THE WORLD: airmail service, £46.99 (6mo); £84.99 (12mo); £164.99 (2yrs). Make payments to 'Practical Electronics', Practical Electronics Subscriptions, PO Box 6337, Bournemouth BH1 9EH United Kingdom. Email: pesubs@selectps.com. PRACTICAL ELECTRONICS is sold subject to the conditions that it shall not, without the written consent of the Publishers first having been given, be lent, resold, hired out or otherwise disposed of by way of Trade at more than the recommended selling price on the cover, and that it shall not be lent, resold, hired out or otherwise disposed of in a mutilated condition or in any unauthorised cover by way of Trade or affixed to or as part of any publication or advertising, literary or pictorial matter.

Did you know our online shop now sells the current issue of *PE* for £5.99 inc. p&p?

Practical Electronics
The UK's premier electronics, computing and maker magazine
Max's Cool Beans
Making automata & building the VFD clock
Teach-In 12
Part 11: long-distance radio communications
Audio Out
Visiting Retro-Tech 2026
Circuit Surgery
Measuring resistance with a 555 timer chip
Mic the Mouse:
a sound-sensitive pet that won't make a mess
Building the USB-C Power Monitor
Home Automation Satellite
Connect sensors, relays and more to a home automation system
PLUS!
Home Automation – running HomeAssistant on a Raspberry Pi
Power Electronics part six – how DC to AC converters work
Techno Talk – display technology changes over the last 50 years
Sep 2026 £5.99
www.electronpublishing.com
@practicalelec
practicalelectronics
Practical Electronics
Audio Out
Linear-phase crossovers
Circuit Surgery
Anti-aliasing and reconstruction filters
Measure USB voltage, current & power up to 60V/±5A
Aug 2026 £5.99
www.electronpublishing.com
@practicalelec
practicalelectronics

You read that right! We now sell the current issue of your favourite electronics magazine for exactly the same price as in the High Street, but we deliver it straight to your door – and for UK addresses we pay the postage. No need to journey into town to queue outside the newsagent. Just go to our website, set up an account in 30 seconds, order your magazine and we'll do the rest.

www.electronpublishing.com

Introducing MPLAB® Code Configurator in VS Code®

Simplify Embedded Development With the MPLAB Code Configurator Extension

Are you an embedded developer using Microsoft® Visual Studio Code® (VS Code®)? The MPLAB® Code Configurator (MCC) that you know and love is now available as an extension for VS Code. This powerful tool is designed to streamline your development process, making it easier than ever to create robust embedded applications.

The MPLAB Code Configurator extension offers a free graphical programming environment that generates seamless, easy-to-understand C code, ready to be integrated into your projects. With an intuitive interface, this extension also enables and configures a rich set of application-specific peripherals and functions, allowing you to focus on what matters most.

Key Features

- Graphical Programming Environment: Simplifies code generation with a user-friendly interface.
- Seamless C Code Generation: Produces easy-to-understand C code that integrates smoothly into your projects.
- Rich Peripheral Configuration: Enables and configures a wide range of application-specific peripherals and functions.
- Intuitive Interface: Designed to be easy to use, reducing the learning curve and speeding up development time.
- Free Extension: Available at no cost, making it accessible for all developers.

Simplify your embedded development today with the MPLAB Code Configurator extension in VS Code. Visit the Extensions Marketplace to download and get started, or learn more about our current and future MPLAB Extensions for VS Code at Microchip.com/MPLABExtensions.

MPLAB® CODE CONFIGURATOR

MPLAB® EXTENSIONS

MICROCHIP

microchip.com/MPLABExtensions

The Microchip name and logo and the Microchip logo are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries. All other trademarks are the property of their registered owners.
© 2026 Microchip Technology Inc. All rights reserved.
MEC2656A-UK-07-26